



TUGAS AKHIR - TE 141599

**RANCANG BANGUN SISTEM PERPESANAN *OBU*
(*ON BOARD UNIT*) UNTUK *INTELLIGENT*
TRANSPORT SYSTEM DI SURABAYA**

MUHAMMAD ALFIAN YASIR
NRP 0711134000083

Dosen Pembimbing
Dr. Ir. Achmad Affandi, DEA.
Ir. Gatot Kusrahardjo, M.T

DEPARTEMEN TEKNIK ELEKTRO
Fakultas Teknologi Elektro
Institut Teknologi Sepuluh Nopember
Surabaya 2018



FINAL PROJECT - TE 141599

***DESIGN OBU (ON BOARD UNIT) MASSAGING
SYSTEM FOR INTELLIGENT TRANSPORT SYSTEM
IN SURABAYA***

**MUHAMMAD ALFIAN YASIR
NRP 07111340000083**

**Lecture Advisor
Dr. Ir. Achmad Affandi, DEA.
Ir. Gatot Kusrahardjo, M.T**

**DEPARTMENT OF ELECTRICAL ENGINEERING
Faculty of Electrical Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2018**

Pernyataan Keaslian Tugas Akhir

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan tugas akhir saya dengan judul “*Rancang Bangun Sistem Perpesanan OBU (On Board Unit) untuk Intelligent Transport System di Surabaya*” adalah benar – benar hasil karya yang dikerjakan secara mandiri, diselesaikan tanpa menggunakan bahan – bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun diambil telah ditulis secara lengkap pada daftar pustaka. Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, Desember 2017

Muhammad Alfian Yasir
2213100083

[Halaman ini sengaja dikosongkan]

**RANCANG BANGUN SISTEM PERPESANAN OBU
(ON BOARD UNIT) UNTUK INTELLIGENT
TRANSPORT SYSTEM DI SURABAYA**

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik**

**Pada
Bidang Studi Telekomunikasi Multimedia
Departemen Teknik Elektro
Institut Teknologi Sepuluh Nopember**

Menyetujui,

Dosen Pembimbing I

Dosen Pembimbing II

Dr. Ir. Achmad Affandi, DEA
NIP. 196510141990021001

Ir. Gatot Kusrahadio, M.T
NIP. 195506221987011001



[Halaman ini sengaja dikosongkan]

ABSTRAK

Intelligent Transport System (ITS) diharapkan dapat mengoptimalkan kondisi angkutan umum saat ini yang belum terintegrasi melalui jaringan TIK serta dapat menanggulangi kemacetan yang selama ini menjadi permasalahan di kota-kota besar khususnya Surabaya. Salah satunya adalah dengan penggunaan *On Board Unit (OBU)* yang ditempatkan pada setiap armada Angkutan Masal Cepat yang diterapkan di kota Surabaya. OBU sendiri merupakan sebuah sistem ITS yang diletakkan di armada, berfungsi sebagai sebuah sistem pengendali masukan dan keluaran yang terkait fungsional manajemen armada, pendapatan (tiket), lalu lintas dan sistem darurat. Tugas akhir ini bertujuan untuk merancang serta membuat sebuah *prototype* sistem yang digunakan untuk proses komunikasi yang terintegrasi antara OBU (*client*) dengan CCROOM (*server*).

Pada pengerjaan tugas akhir ini dilakukan dua tahap pada proses perancangan dan implementasi, yaitu pertama membuat arsitektur jaringan keseluruhan dan aplikasi sistem yang di dalamnya terdapat *fitur-fitur* pendukung untuk digunakan pada sistem komunikasi angkutan massal cepat (AMC). Aplikasi pengiriman data dari *client-server* dibuat menggunakan *software Borland Delphi 6*. Jaringan yang digunakan adalah jaringan seluler dengan menggunakan provider Tri sebagai paket data, pengiriman informasi dari armada melalui jaringan seluler mengirimkan paket data menuju tower terdekat dan di alirkan ke *router* hingga di terima di *server*.

Pengujian sistem komunikasi ini dilakukan dengan 2 macam pengujian, yaitu pengujian sistem *client-server* beserta *fitur-fitur* yang ada di dalamnya seperti *e-ticketing*, *emergency button*, *login* *logout* serta pengujian kualitas jaringan pada saat sistem sedang berjalan.

Kata kunci: *Intelligent Transportation System*, *e-ticketing*, *emergency button*, *login* *logout*

[Halaman ini sengaja dikosongkan]

ABSTRACT

Intelligent Transport system (ITS) is expected to optimized the public transportation condition which right now is not integrated yet through ICT network, also to overcome the traffic jam that has been a problem in big cities especially Surabaya. One of them is the use of OBU which is placed in every Fast Mass Transportation vehicle which is applied in Surabaya. OBU it self is a ITS system that is placed in the vehicle, functioning as an input and output controlling system related to vehicle fungsional manangement, income (ticket), traffic and emergency system.

This thesis purpose is to design and make the system prototype that can be use for communication process whoch ksnintegrated between OBU (client) with CCROOM (server) In the making of this thesis, 2 step as done in the designing and implementation process, first is the making of all network architecture and system application which includes supporting features to be use in the AMC communication. Data sending application from client-server is made using Borland Delphi 6 *software*. The network used is a cellular network using Tri provider as a data packet, sending information from fleets over cellular networks to send data packets to the nearest tower and routed to the router until received on the server

The testing of this communication system is done by 2 types of testing, which is client-server system testing as well as the features inside it like *e-ticketing*, emergency buton, login logout aswell as network quality testing when the system is working

Keywords: *Intelligent Transportation System, e-ticketing, emergency button, login logout*

[Halaman ini sengaja dikosongkan]

KATA PENGANTAR

Segala Puji bagi Allah SWT, karena atas berkat dan rahmat-Nya serta kasih sayang yang diberikan NYA, saya dapat menjalani dan menyelesaikan tugas akhir yang berjudul “**Rancang Bangun Sistem Perpesanan OBU (On Board Unit) untuk Intelligent Transport System di Surabaya**” Tugas akhir ini merupakan salah satu syarat yang harus dipenuhi dalam menyelesaikan program studi strata 1 (S1) pada Departemen Teknik Elektro, Fakultas Teknologi Elektro, Institut Teknologi Sepuluh Nopember Surabaya.

Terima kasih saya ucapkan kepada orang tua serta kedua adik saya yang selalu mendoakan dan memberikan segala dukungan. Kemudian, kepada Bapak Dr. Ir. Achmad Affandi, Ir. Gatot Kusrahardjo, M.T , dan Bapak Didit selaku dosen pembimbing yang selalu memberi bimbingan dan masukan demi terselesaikannya tugas akhir ini. Terimakasih juga tidak lupa saya sampaikan untuk dosen-dosen ITS khususnya Teknik Elektro yang telah memberikan segala ilmu dalam proses belajar mengajar, semoga ilmu yang diberikan akan senantiasa bermanfaat bagi penulis dan lingkungan sekitar penulis. Kemudian terima kasih kepada Ahmad Nur Fikri sebagai mentor yang luar biasa dalam pengerjaan tugas akhir ini, teman-teman seperjuangan Teknik Elektro ITS 2013 E-53 yang senantiasa memberikan bantuan baik melalui diskusi maupun semangat, serta Siti Sholihah Putri sebagai pendamping terbaik dalam hidup ini dan selama proses pengerjaan Tugas Akhir.

Hasil dan manfaat dari penelitian ini diharapkan dapat digunakan untuk meningkatkan kualitas layanan transportasi umum (AMC) yang berbasis ITS khususnya di kota Surabaya. Dalam proses perbaikan dan pengembangan, kritik dan saran terhadap penelitian ini sangat dibutuhkan untuk mengetahui kekurangan dan kebutuhannya. Terima kasih.

Surabaya, Desember 2017

penulis

[Halaman ini sengaja dikosongkan]

DAFTAR ISI

ABSTRAK	i
ABSTRACT	iii
KATA PENGANTAR	v
DAFTAR ISI	vii
DAFTAR GAMBAR	xi
DAFTAR TABEL	xv
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah.....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan	2
1.5 Metodologi	3
1.6 Sistematika Penulisan.....	4
1.7 Manfaat	4
BAB II TINJAUAN PUSTAKA	7
2.1 Smart City	7
2.2 <i>Intelligent Transportation System (ITS)</i>	8
2.2.1 Advanced Traveler Information System (ATIS)	9
2.2.2 Advanced Transportation Management Systems (ATMS)	10
2.2.3 ITS-Enabled Transportation Pricing Systems (ITS-ETPS)...	10
2.2.4 Advanced Public Transportation Systems (APTS).....	11
2.2.5 Fully integrated ITS, termasuk integrasi vehicle-to- infrastructure (VII) and vehicle-to-vehicle (V2V)	12

2.3 <i>Client-Server</i>	13
2.3.1 Jaringan Seluler	13
2.3.2 Socket	14
2.4 <i>On Board Unit (OBU)</i>	14
2.5 Variable messaging system (VMS)	15
2.6 Quality of Service (QOS)	17
2.6.1 Throughput	17
2.6.2 <i>Delay</i>	17
2.7 Angkutan Masa Cepat (AMC)	18
2.7.1 Sistem Jaringan Komunikasi	18
BAB III PERANCANGAN DAN IMPLEMENTASI SISTEM	21
3.1 Perancangan Sistem Komunikasi Terintegrasi	21
3.1.1 Perangkat <i>Server</i> yang Digunakan	22
3.1.2 Perancangan Aplikasi <i>On Board Unit (OBU)</i>	22
3.1.3 Perancangan <i>Database</i> dan <i>Web Report</i>	25
3.2 Perancangan Perpesanan Penumpang	26
3.3 Perancangan Perpesanan <i>Driver</i>	28
3.4 Perancangan <i>Fitur Ticketing</i>	31
3.5 Perancangan Sumber Tenaga pada Armada	32
3.6 Implementasi Sistem Komunikasi Terintegrasi	33
3.6.1 Implementasi Sistem	33
3.6.2 Implementasi Jaringan	35
3.7 Skenario Pengujian Sistem	35
3.7.1 Skenario Pengujian Sistem	37

3.7.2 Skenario Pengujian Jaringan.....	38
BAB IV PENGUJIAN DAN ANALISIS SISTEM	41
4.1 Pengujian Sistem pada <i>OBU side</i>	41
4.1.1 Pengujian <i>OBU</i> untuk Pemantauan Armada	44
4.1.2 Pengujian <i>VMS</i> Penumpang	46
4.1.3 Pengujian <i>VMS Driver</i>	47
4.1.4 Pengujian <i>Fitur Ticketing</i>	50
4.1.5 Analisa Hasil Pengujian Sistem pada <i>OBU Side</i>	52
4.2 Pengujian Sistem pada Halte <i>Side</i>	53
4.2.1 Analisa Hasil Pengujian Sistem pada Halte <i>Side</i>	54
4.3 Pengujian dan Analisa Jaringan Komunikasi Sistem	55
4.3.1 Pengujian dan Analisa <i>Throughput</i>	55
4.3.2 Pengujian dan Analisa <i>Delay</i>	59
BAB V PENUTUP	63
5.1 Kesimpulan	63
5.2 Saran.....	63
DAFTAR PUSTAKA	65
LAMPIRAN	67
RIWAYAT HIDUP.....	93

[Halaman ini sengaja dikosongkan]

DAFTAR GAMBAR

Gambar 2.1 Komponen ITS	8
Gambar 2.2 Kategori ITS	9
Gambar 2.3 Penerapan Teknologi ATMS	10
Gambar 2.4 Perangkat ETC dan Kartu Pembayaran Otomatis	11
Gambar 2.5 Informasi Kedatangan dan Status Transportasi Publik	12
Gambar 2.6 Arsitektur <i>Client-Server</i>	13
Gambar 2.7 Arsitektur Jaringan <i>OBU</i>	15
Gambar 2.8 Penerapan <i>VMS</i> dalam Teknologi ATIS	16
Gambar 2.9 Desain Manajemen Armada <i>AMC</i>	19
Gambar 3.1 Arsitektur Perpesanan	21
Gambar 3.2 <i>Flowchart</i> Aplikasi <i>OBU</i>	23
Gambar 3.3 Tampilan Aplikasi Delphi 6	24
Gambar 3.4 Tampilan Tabel <i>Database</i> di <i>Server</i>	25
Gambar 3.5 Tampilan <i>Web Report</i>	26
Gambar 3.6 <i>Flowchart</i> Perpesanan Penumpang	27
Gambar 3.7 <i>Flowchart</i> Informasi <i>VMS</i> Penumpang	27
Gambar 3.8 <i>Flowchart Button Notification</i>	28
Gambar 3.9 <i>Flowchart Absensi Driver</i>	29
Gambar 3.10 Aliran Informasi Absensi <i>Driver</i>	29
Gambar 3.11 <i>Flowchart Emergency Button</i>	30
Gambar 3.12 <i>Flowchart</i> Registrasi Member Baru	31
Gambar 3.13 <i>Flowchart</i> Top up Saldo	32
Gambar 3.14 Rangkaian Box Aki	33
Gambar 3.15 <i>Flowchart</i> Implementasi Sistem	34
Gambar 3.16 Implementasi Sistem Jaringan	35

Gambar 3.17	<i>Flowchart</i> Pengujian <i>Client-server</i>	37
Gambar 3.18	Proses Informasi dari Armada ke Server	38
Gambar 3.19	<i>Flowchart</i> Pengujian Jaringan	39
Gambar 4.1	<i>Point to Point</i> dari Titik Kordinat	43
Gambar 4.2	Pengukuran Jarak Tespoint 1 dengan Lainnya	43
Gambar 4.3	Jarak Tespoint.....	43
Gambar 4.4	Tampilan Awal Aplikasi	44
Gambar 4.5	Tampilan Hasil <i>Messaging</i>	45
Gambar 4.6	Hasil <i>Messaging</i> disimpan dalam <i>Database</i>	45
Gambar 4.7	Hasil <i>Messaging</i> pada <i>Web Report</i>	46
Gambar 4.8	Tampilan Awal Aplikasi & Hasil pada Monitor.....	46
Gambar 4.9	Tampilan Absen <i>Driver</i> Sekaligus <i>Login</i> Aplikasi.....	47
Gambar 4.10	Hasil Absensi di <i>Database</i>	48
Gambar 4.11	Hasil pada <i>Web Report</i>	48
Gambar 4.12	Tampilan <i>Emergency Button</i>	49
Gambar 4.13	<i>Report Emergency Button</i> di <i>Database</i>	49
Gambar 4.14	Tampilan Registrasi Member	50
Gambar 4.15	<i>Report</i> Registrasi Member Baru	51
Gambar 4.16	Hasil Tambah Saldo	51
Gambar 4.17	Halte Asrama ITS	53
Gambar 4.18	Halte Robotika.....	54
Gambar 4.19	Halte Teknik Lingkungan.....	54
Gambar 4.20	Cek IP dengan <i>Command Prompt</i>	56
Gambar 4.21	Ping Server Menggunakan <i>Command Prompt</i>	56
Gambar 4.22	Tampilan <i>Capture</i> di <i>Wireshark</i>	57
Gambar 4.23	Tampilan <i>Save CSV</i> pada <i>Wireshark</i>	57
Gambar 4.24	Grafik Pengujian <i>Throughput</i>	58

Gambar 4.25 Ping Server Menggunakan Command Prompt	59
Gambar 4.26 Tampilan Data di <i>Wireshark</i>	60
Gambar 4.27 Grafik Pengujian <i>Delay</i>	60

[Halaman ini sengaja dikosongkan]

DAFTAR TABEL

Tabel 2.1 Standarisasi <i>Delay</i> ITU-T	18
Tabel 3.1 Syarat Perancangan Sistem	36
Tabel 4.1 Hasil Pengukuran Modul GPS	42
Tabel 4.2 Hasil Pengujian Sistem pada <i>OBU Side</i>	52
Tabel 4.3 Hasil Pengujian Sistem pada Halte <i>Side</i>	55

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Teknologi informasi dan komunikasi (TIK) telah membawa perubahan mulai dari bidang industri, pendidikan, kesehatan, pemerintahan dan hingga ke bidang transportasi secara perlahan dan bertahap. Di bidang transportasi, kehadiran TIK diharapkan memberikan keefesiensian pada operasional, keefektifan pelayanan kepada masyarakat dan dapat memberikan kelancaran pada sistem transportasi secara umum. Surabaya telah mencanangkan sejak tahun 2005, dengan penerapan *Automatic Traffic Control System* (ATCS) sebagai Cyber City, dan terus berusaha menjadi kota yang unggul di bidang penerapan teknologi informasi dan komunikasi (TIK).

Setiap warga di Indonesia pasti memiliki keinginan untuk menjadikan kota tempat mereka tinggal sebagai kota percontohan yang menuai banyak manfaat bagi warga di dalamnya dan memberikan dampak positif kepada kota lainnya. Kota Surabaya yang merupakan sebuah kota besar yang berada di Indonesia mengalami jumlah kenaikan kendaraan pribadi setiap tahunnya sehingga dapat memberikan dampak negatif bagi lingkungan maupun bagi lalu lintas di Surabaya. Dampak negatif yang ditimbulkan merupakan penyebab dari kurangnya transportasi publik yang memadai di kota Surabaya.

Untuk mewujudkan manajemen armada yang bekerja dengan baik, Angkutan Massal Cepat (AMC) memiliki sistem komunikasi terintegrasi dengan dipasangkan *On-Board Unit* (OBU) pada armada dan halte. Data yang diperoleh dari OBU akan diteruskan melalui jaringan seluler, kemudian seluruh data yang di dapat akan diterima oleh *Server TMC Control Center Room* (CC-ROOM) untuk selanjutnya informasi akan difilter agar tepat sasaran. Maka dibuat lah aplikasi serta *prototype* yang mampu mensimulasikan dari tiap OBU system untuk memudahkan pengiriman informasi kepada calon penumpang agar mendapat jadwal secara *real time* tanpa menunggu dalam waktu yg lama.

1.2 Perumusan Masalah

Permasalahan yang dibahas dalam Tugas Akhir ini adalah :

1. Bagaimana rancangan dan penerapan *fitur-fitur* pada aplikasi *On-Board Unit (OBU)* untuk merekam kejadian pada Angkutan Massa Cepat (AMC) yaitu posisi, waktu, identitas armada?
2. Bagaimana melakukan pemindahan informasi serta mendapatkan nilai *Throughput* dan *Delay* dari sistem yang telah dirancang?

1.3 Batasan Masalah

Hal-hal yang akan dilakukan dalam penelitian ini adalah sebagai berikut:

1. Aplikasi yang dibuat berfungsi sebagai *Client-Server*
2. Perangkat lunak yang akan digunakan untuk membangun aplikasi ini adalah Borland Delphi 6
3. Pengujian pengiriman paket data akan dilakukan dengan menggunakan jaringan seluler.
4. *Throughput* terbaik
5. *Delay* yang rendah

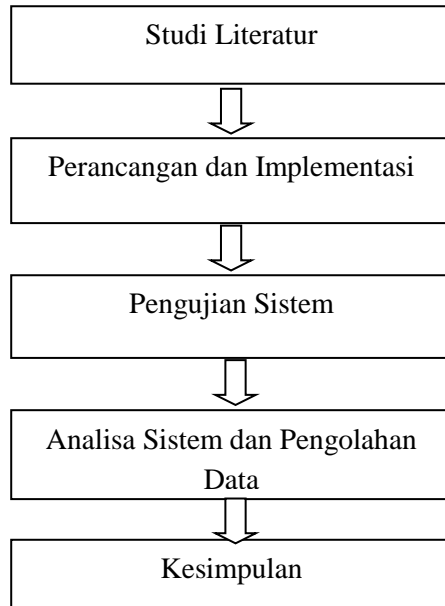
1.4 Tujuan

Tujuan dari Tugas Akhir ini diharapkan sebagai berikut :

1. Membangun sistem informasi transportasi pintar yang dapat melakukan pemantauan terhadap armada menggunakan GPS-*OBU* yang terintegrasi dengan *Control Center Room (CC-ROOM)* sebagai *server* atau media penyimpanan segala informasi.
2. Dapat memberikan informasi layanan secara *real time* untuk membantu penumpang dan calon penumpang agar mendapatkan informasi yang akurat tentang penjadwalan, posisi keberadaan armada, serta tujuan selanjutnya.
3. Membangun prototype *OBU* dan *fitur VMS* yang dapat diterapkan pada armada angkutan umum berbasis sistem transportasi cerdas.
4. Mendapatkan nilai *Throughput* dan *delay* dalam proses pengiriman informasi dari *OBU* menuju *server*.

1.5 Metodologi

Berikut adalah tahapan-tahapan yang dilalui dalam pengerjaan tugas akhir ini sebagai berikut:



1. Studi Literatur
Pada tahap ini akan dilakukan pemahaman serta pengambilan data segala macam teori yang ada di buku, paper, maupun referensi yang ada untuk kebutuhan pengerjaan Tugas Akhir ini.
2. Perancangan dan Implementasi Sistem
Pada tahapan ini akan membuat sebuah rancangan sebuah sistem yang terintegrasi
3. Pengujian Sistem
Setelah melakukan perancangan sistem yang akan dibuat pada perangkat lunak akan diuji coba sistem tersebut dan mengumpulkan data-data yang dibutuhkan

4. Analisa dan Pengolahan Data

Setelah mendapatkan data dari skema pengujian, maka akan dilakukan perbandingan dan pengolahan data dari hasil data tersebut. Sehingga nantinya data tersebut bisa dianalisa dan diambil mana yang lebih efisien serta akan menghasilkan kesimpulan dari penelitian tugas akhir ini.

1.6 Sistematika Penulisan

Sistematika penulisan pada Tugas Akhir ini dibagi menjadi beberapa bab dengan rincian:

BAB I PENDAHULUAN

Menguraikan latar belakang, perumusan masalah, batasan masalah, tujuan dan manfaat yang berkaitan dengan pengerjaan dan penyusunan Tugas Akhir ini.

BAB II TINJAUAN PUSTAKA

Pada bab ini akan dibahas landasan teori mengenai konsep *Smart City*, *Intelligent Transportation System*, *Surabaya Mass Rapid Transportation*, sistem komunikasi *Client-Server*, *On Board Unit (OBU)*, *Quality of Service (QOS)*, protokol komunikasi untuk pengiriman paket data dari *Client-Server*, dan *Variable messaging system (VMS)*.

BAB III PERANCANGAN DAN IMPLEMENTASI SISTEM

Pada Bab ini berisikan perancangan data base, kemudian aplikasi sistem yang akan digunakan berbasis Delphi 6, serta diimplementasi sistem yang telah dibuat apakah sistem berjalan dengan baik atau tidak.

BAB IV PENGUJIAN DAN ANALISIS SISTEM

Pada bab ini berisi hasil dari rancangan yang sudah dibuat pada bab III. Dari pengujian ini kemudian dianalisis dan ditarik kesimpulan sementara mengenai parameter-parameter yang telah diuji

BAB V PENUTUP

Pada bab ini berisi kesimpulan dari keseluruhan materi dan dari hasil analisis data pada bab IV. Selain itu pada bab ini dibahas mengenai saran yang bisa dilakukan untuk pengembangan penelitian selanjutnya

1.7 Manfaat

Dari tujuan tersebut diharapkan mampu meminimalkan waktu tunggu bagi calon penumpang angkutan umum, serta mampu memberikan dampak positif bagi sistem transportasi yang ada di Surabaya. Hasil dari Tugas Akhir ini diharapkan dapat memberikan manfaat sebagai berikut :

1. Diharapkan dapat menjadi acuan untuk perkembangan transportasi di Indonesia khususnya pada sistem pemantauan armada serta *messaging system*.
2. Memberikan kontribusi real terhadap sistem transportasi di Surabaya yang menuju kota percontohan bagi kota lainnya di Indonesia maupun dunia.
3. Membangun sistem manajemen lalu lintas dan perjalanan armada AMC yang terintegrasi dengan *OBU* dan *VMS*.
4. Dapat membantu serta meningkatkan kualitas kinerja layanan, operasional, maupun pandangan publik terhadap angkutan umum.

[Halaman ini sengaja dikosongkan]

BAB II

TINJAUAN PUSTAKA

2.1 Smart City

Merupakan sebuah cita-cita yang pasti diinginkan oleh seluruh kota besar di Indonesia maupun dunia, Perencanaan *smart city* merupakan sebuah agenda besar yang dipersiapkan oleh seluruh kota besar sebagai respon konseptual dan praktis terhadap berbagai krisis perkotaan di dunia yang semakin mengawatirkan, untuk mengembalikan hubungan antara manusia, ruang binaan dan ruang alami yang lebih harmonis, sehingga tidak adanya ketimpangan bagi tiap mahluk yang ada di dalamnya. Smart city adalah sebuah konsep kota cerdas/pintar yang membantu masyarakat yang berada di dalamnya untuk mengelola sumber daya yang dimiliki secara efisien dan dapat memberikan informasi yang tepat kepada masyarakat/lembaga dalam melakukan kegiatannya ataupun mengantisipasi kejadian yang tak terduga sebelumnya. Dalam penerapan *smart city* ini sendiri, ada beberapa klasifikasi yang diberikan di dalamnya yaitu:

- ***Smart Government (pemerintahan pintar)***, kunci utama keberhasilan penyelenggaraan pemerintahan adalah *Good Governance*, untuk mencapai hal tersebut perlu memerhatikan aspek yang ada di dalamnya. Yaitu pandangan publik, sistem dan proses penyelenggaraan pemerintahan serta pembangunan yang mengindahkan prinsip-prinsip supremasi hukum.
- ***Smart Economy (Ekonomi pintar)***, maksud dari ekonomi pintar ini adalah semakin tinggi inovasi-inovasi baru yang di buat serta ditingkatkan maka akan menambah peluang usaha baru dan meningkatkan persaingan pasar usaha/modal.
- ***Smart Mobility (mobilitas pintar)***, pengelolaan infrastruktur kota yang dikembangkan di masa depan merupakan sebuah sistem pengelolaan terpadu untuk menjamin keberpihakan pada kepentingan publik.
- ***Smart People (orang/masyarakat pintar)***, menjadikan sebuah kota pintar pastinya membutuhkan modal yang sangat besar, baik modal ekonomi, modal sumber daya manusia maupun modal sosial.
- ***Smart Living (Lingkungan pintar)***, lingkungan pintar itu berarti lingkungan yang bisa memberikan kenyamanan,

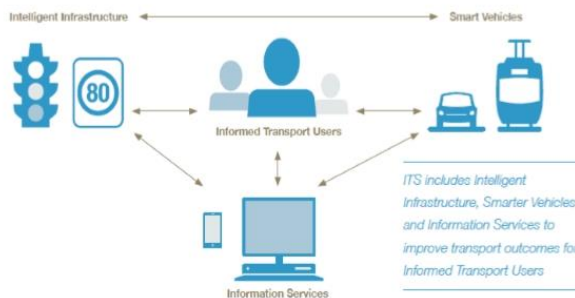
keberlanjutan sumber daya, keindahan fisik maupun non fisik, visual maupun tidak, bagi masyarakat dan publik.

- **Smart Live (Hidup pintar)**, kehidupan berbudaya dan berkarakter sehingga manusia yang ada di dalamnya memiliki kualitas hidup yang terukur (budaya).

Konsep *smart city* ini kini menjadi impian banyak kota besar di Indonesia. Konsep *smart city* dianggap sebagai solusi dalam mengatasi berbagai macam aspek masalah yang ada di tiap kota seperti kemacetan yang merayap, sampah yang berserakan, ataupun pemantau kondisi lingkungan di suatu tempat. Perjalanan menuju konsep *smart city* ini juga sudah mulai berjalan pelan-pelan di tiap kota di Indonesia. Dukungan aplikasi serta teknologi yang terus berkembang pada zaman sekarang serta terciptanya ekosistem kreatif di bidang teknologi, merupakan langkah awal yang akan direalisasikan menuju kota pintar. [5]

2.2 Intelligent Transportation System (ITS)

Intelligent Transport System merupakan sistem transportasi cerdas yang menerapkan teknologi informasi dan komunikasi yang mendukung dan mengoptimalkan semua model transportasi dengan meningkatkan efektivitas biaya, bagaimana mereka bekerja, baik secara individu maupun bekerja sama satu sama lain.



Gambar 2.1 Komponen ITS

Seperti gambar 2.1 bahwa sistem cerdas dibagi menjadi tiga bidang yaitu:

1. Infrastruktur cerdas (*intelligent infrastructure*), contohnya sinyal lalu lintas di jalan raya, tanda pesan variable untuk mengingatkan pengguna jalan dari bahaya yang terjadi di depan, dan sinyal *ramp* jalan bebas hambatan yang bekerja untuk menjaga kelancaran jalan raya.
2. Kendaraan cerdas (*smart vehicle*), seperti notifikasi tabrakan otomatis, pemberitahuan batas kecepatan, peringatan tabrakan mudur dan depan, sistem navigasi GPS, dan *interlock*. Layanan informasi (*information services*), contohnya informasi kedatangan bus pada ponsel, sistem navigasi dalam mobil yang menerima kondisi lalu lintas terkini untuk panduan kemacetan, dan program akses nasional untuk truk.[1]

ITS Category	Specific ITS Applications
1. Advanced Traveler Information Systems (ATIS)	Real-time Traffic Information Provision Route Guidance/Navigation Systems Parking Information Roadside Weather Information Systems
2. Advanced Transportation Management Systems (ATMS)	Traffic Operations Centers (TOCs) Adaptive Traffic Signal Control Dynamic Message Signs (or "Variable" Message Signs) Ramp Metering
3. ITS-Enabled Transportation Pricing Systems	Electronic Toll Collection (ETC) Congestion Pricing/Electronic Road Pricing (ERP) Fee-Based Express (HOT) Lanes Vehicle-Miles Traveled (VMT) Usage Fees Variable Parking Fees
4. Advanced Public Transportation Systems (APTS)	Real-time Status Information for Public Transit System (e.g. Bus, Subway, Rail) Automatic Vehicle Location (AVL) Electronic Fare Payment (for example, Smart Cards)
5. Vehicle-to-Infrastructure Integration (VII) and Vehicle-to-Vehicle Integration (V2V)	Cooperative Intersection Collision Avoidance System (CICAS) Intelligent Speed Adaptation (ISA)

Gambar 2.2 Kategori ITS

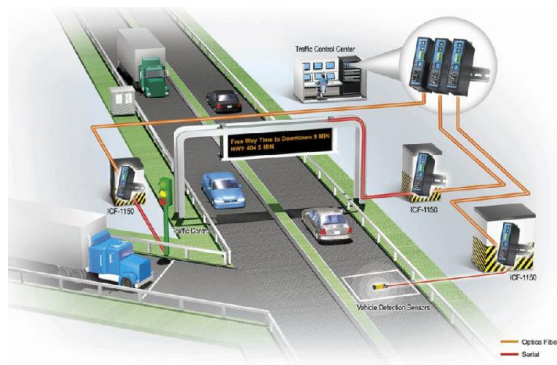
2.2.1 Advanced Traveler Information System (ATIS)

ATIS, sistem informasi canggih bagi pengendara, menyediakan informasi *real-time*, seperti rute transit dan jadwal; arah navigasi; dan informasi tentang penundaan akibat kemacetan, kecelakaan, kondisi cuaca, atau perbaikan jalan kerja Penerapan ITS, sistem informasi perjalanan yang paling efektif dapat menginformasikan *driver* secara *real-time* lokasi tepat mereka, memberitahu mereka tentang lalu lintas, jalan atau kebutuhan lain yang terkait pada saat ini di jalan raya dan sekitarnya, dan memberdayakan secara optimal pemilihan rute dan

instruksi navigasi, idealnya membuat informasi ini tersedia di beberapa platform, baik di dalam maupun di luar kendaraan. Gambar 2.1 menggambarkan tiga aspek kunci untuk penyediaan informasi lalu lintas *real-time* yaitu pengumpulan, pengolahan, dan penyebaran informasi, dengan setiap langkah yang melibatkan satu set yang berbeda dari perangkat teknologi, *platform* dan aktor, baik publik maupun privat.

2.2.2 Advanced Transportation Management Systems (ATMS)

ATMS, sistem manajemen transportasi canggih, meliputi perangkat kontrol lalu lintas, seperti lampu lalu lintas, meter jalan, tanda/pesan status lalu lintas yang dinamis, dan pusat-pusat operasi lalu lintas. *Traffic Operations Centers* (TOCs), pusat operasi lalu lintas, merupakan pusat pengelolaan lalu lintas yang dioperasikan oleh kota mengandalkan teknologi informasi untuk menghubungkan sensor dan perangkat di pinggir jalan, probe kendaraan, kamera, tanda-tanda pesan, dan perangkat lain bersama-sama untuk menciptakan pandangan yang terintegrasi dari aliran trafik dan mendeteksi kecelakaan, peristiwa cuaca yang berbahaya, atau bahaya jalan lainnya.



Gambar 2.3 Penerapan Teknologi ATMS

2.2.3 ITS-Enabled Transportation Pricing Systems (ITS-ETPS)

ETPS, merupakan peran sentral dalam pembiayaan sistem transportasi. Contoh penerapan ETPS termasuk sistem pembayaran

otomatis, misalnya pengumpulan tol elektronik (*Electronic Toll Collection, ETC*), ERP untuk mengurangi kemacetan pada jalan utama, jalur *fee-based express* (*HOT, High Occupancy Toll*), dan sistem pembayaran berbasis jarak (*Vehicle Miles Traveled, VMT*). Pada ETC maupun ERP, pembayaran dapat dilakukan secara otomatis melalui perangkat *DSRC-enabled* atau *tag on-board* yang di tempatkan dikaca depan. Australia dan Jepang, menerapkan standar ETC canggih yang mencakup sistem nasional tunggal. Amerika Serikat, menerapkan beberapa sistem pembiayaan ETC untuk perjalanan lintas negara karena operator ETC yang berbeda, dan belum ada interoperabilitas.



Gambar 2.4 Perangkat ETC dan Kartu Pembayaran Otomatis

Bentuk lain penerapan ETPS, contohnya pada penerapan sistem pembayaran jumlah km perjalanan (*VMT, vehicle miles traveled*) yang dikenakan kepada pengendara untuk setiap meternya. Sistem VMT merupakan salah satu bentuk pajak bahan bakar dan biaya lain yang diterapkan di banyak negara untuk membiayai sistem transportasi. Penerapan “Kilometerprijs” (harga per kilometer) di Belanda merupakan sistem VMT pertama nasional di dunia dilaksanakan untuk baik untuk kendaraan penumpang maupun kendaraan berat.

2.2.4 Advanced Public Transportation Systems (APTS)

Teknologi APTS adalah kumpulan teknologi yang meningkatkan efisiensi dan keselamatan public sistem transportasi dan akses menawarkan pengguna yang lebih besar untuk informasi tentang sistem operasi. Itu implementasi teknologi APTS mengubah cara sistem transportasi umum beroperasi, dan mengubah sifat dari jasa transportasi yang dapat ditawarkan oleh angkutan umum sistem. Tujuannya adalah

untuk menyediakan transportasi umum pembuat keputusan informasi lebih lanjut untuk membuat keputusan yang efektif pada sistem dan operasi serta meningkatkan kenyamanan dan penumpang wisatawan.

APTS, sistem transportasi publik canggih, memungkinkan moda transpor publik seperti kereta, trem dan bus dapat melaporkan posisi mereka (*automatic vehicle location*, AVL), sehingga penumpang mendapat informasi status real-time mereka (informasi kedatangan dan keberangkatan) dan bagi pengelola dapat memonitor keberadaan aset mereka. [2]



Gambar 2.5 Informasi Kedatangan dan Status Transportasi Publik

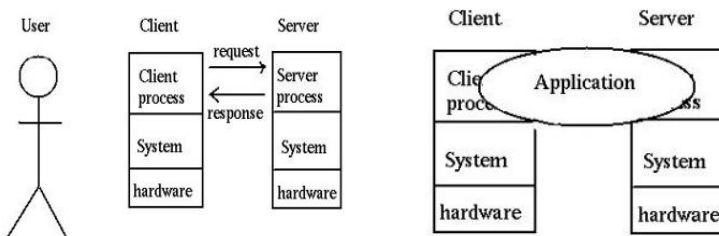
2.2.5 Fully integrated ITS, termasuk integrasi vehicle-to-infrastructure (VII) and vehicle-to-vehicle (V2V)

Terkoneksinya antar elemen dan aset sistem transportasi, seperti semua jenis sensor yang digunakan, lampu lalu lintas, dan kendaraan atau antar moda yang lain. Integrasi VII merupakan pola dasar ITS yang komprehensif. Di Amerika, inisiatif VII dikenal IntelliDriveSM, 2009, diimplementasikan dan merupakan infrastruktur komunikasi yang mendukung VII maupun V2V. Juga komunikasi untuk berbagai aplikasi keselamatan dan operasi transportasi. IntelliDrive memungkinkan tag atau sensor DSRC-enabled, jika banyak digunakan di kendaraan, jalan raya, dan di pinggir jalan atau peralatan persimpangan, akan memungkinkan elemen inti dari sistem transportasi cerdas berkomunikasi dengan satu sama lain, akan memberikan berbagai manfaat.

2.3 Client-Server

Client-Server adalah arsitektur jaringan yang memisahkan *Client* (biasanya aplikasi yang menggunakan GUI) dengan *Server*. Masing-masing *Client* dapat meminta data atau informasi dari *Server* kemudian Sistem *Client-Server* didefinisikan sebagai sistem terdistribusi, di mana yang satu akan bertindak sebagai *Server* atau pemberi layanan dan yang lainnya sebagai *Client* atau peminta layanan. Baik *Client* maupun *Server* memiliki pemroses atau CPU sendiri sedangkan jaringan yang digunakan bisa berupa jaringan lokal (LAN) ataupun jaringan yang lebih luas (WAN).

Client adalah komputer yang meminta (request) satu layanan tertentu ke suatu *Server*. Komputer *Client* harus dilengkapi dengan aplikasi *Client* khusus dan menjalankannya, sehingga dapat memanfaatkan layanan yang ditawarkan *Server*. Selain *Client*, *Server* adalah komputer yang menawarkan suatu layanan tertentu kepada komputer atau jaringan lain.



Gambar 2.6 Arsitektur *Client-Server*

2.3.1 Jaringan Seluler

Dalam aplikasi yang diterapkan pada *Intelligent Transport System* (ITS) salah satu komponen yang penting adalah sebuah Jaringan, Karena Jaringan seluler menjangkau area yang jauh lebih luas, Jangkauan umumnya mencakup nasional dengan infrastruktur jaringan *wireless* yang disediakan oleh *wireless service carrier* untuk biaya pemakaian bulanan, mirip dengan langganan ponsel yang menggunakan macam-macam sistem transmisi mulai dari CDMA, GSM, EDGE, 3G, dan HSPDA.

Jaringan *wireless* beroperasi dalam sebuah jaringan yang membagi kota atau wilayah kedalam sel-sel yang lebih kecil. Satu sel mencakup beberapa blok kota atau sampai 250 mil persegi. Setiap sel menggunakan sekumpulan frekuensi radio atau saluran-saluran untuk

memberikan layanan di area spesifik. Kekuatan radio ini harus dikontrol untuk membatasi jangkauan sinyal geografis. Oleh Karena itu, frekuensi yang sama dapat digunakan kembali di sel terdekat. Maka banyak orang dapat melakukan percakapan secara simultan dalam sel yang berbeda di seluruh kota atau wilayah, meskipun mereka berada dalam satu saluran.

Dalam setiap sel, terdapat stasiun dasar yang berisi antenna *wireless* dan perlengkapan radio lain. Antenna *wireless* dalam setiap sel akan menghubungkan sumber ke jaringan telepon *local*, internet, ataupun jaringan *wireless* lain. Antenna *wireless* mentransmisikan sinyal.

2.3.2 Socket

Struktur *Client-Server* memungkinkan untuk berbagi informasi dan sumber daya antar sistem, misalnya file, *disk space*, prosesor, dan *peripheral* dapat berkolaborasi dan menyampaikan pesan pasti antara prosesor. Beberapa prosesor dijalankan sebagai *Client* dan ada yang sebagai *Server*. Harus ada semacam mekanisme yang membuat setiap prosesor dapat mengetahui proses berjalannya jaringan mesin lain ketika dua prosesor sedang berkomunikasi melalui jaringan.

Socket menyediakan *interface* pemrograman sederhana yang memiliki hubungan erat dengan proses komunikasi single dan multi mesin. Tujuan dari *socket* adalah untuk menyediakan sebuah metode umum IPC pada *transport layer*, apakah prosesornya berjalan pada mesin yang sama.

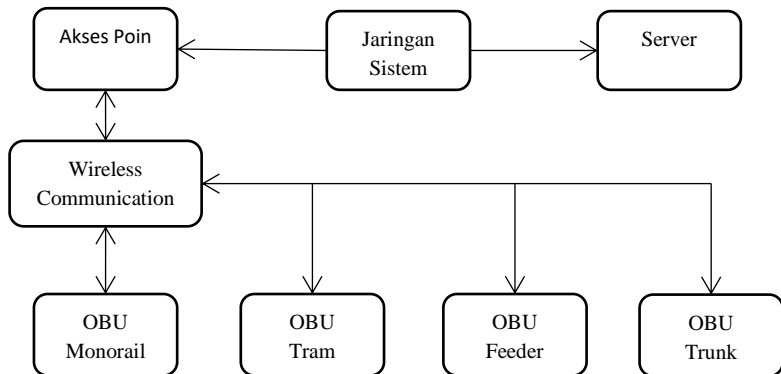
2.4 On Board Unit (OBU)

Penerapan akuisisi ID pengemudi dan penumpang dilakukan oleh salah satu fungsi *OBU* melalui antar muka sistem *e-ticketing*. Rancangan sistem tiket elektronik diawali dengan mengakuisisi data penumpang melalui e-tiket dalam bentuk RFID oleh *OBU*. Jadi *OBU* adalah sistem ITS yang diletakkan di armada, berfungsi sebagai sebuah sistem pengendali masukan dan keluaran yang terkait fungsional manajemen armada, pendapatan (tiket), lalu lintas dan sistem darurat. Sehingga *OBU* akan dirancang mendukung beberapa fungsi berikut: [14]

- Inisiasi dan penutupan sesi perjalanan armada AMC oleh pengemudi
- Identifikasi pengemudi dan penumpang, catuan ke sistem *e-ticketing*
- Pemantauan posisi armada AMC
- Inisiasi sistem emergency armada

- Mendukung sistem *VMS* armada (display informasi perjalanan bagi penumpang armada AMC)
- Mendukung integrasi dengan sistem manajemen lalu lintas (*Traffic Management Center*) dan perjalanan armada AMC termasuk ke sistem *VMS*, sistem pendapatan (*e-ticketing*).

Pada AMC Surabaya, akses poin Wi-Fi akan terdapat pada lokasi koridor pemberhentian. Akses poin tersebut direncanakan untuk membantu pengiriman data melalui jaringan internet. Hal tersebut berarti *OBU* yang digunakan harus memiliki kemampuan untuk mengakses Wi-Fi. Model pengiriman data ke *Server* akan lebih mudah dan dapat dilakukan seketika serta lebih hemat energi. Gambar 2.7 menunjukkan diagram blok rancangan arsitektur sistem jaringan *OBU* dalam kerangka sistem ITS di Surabaya.

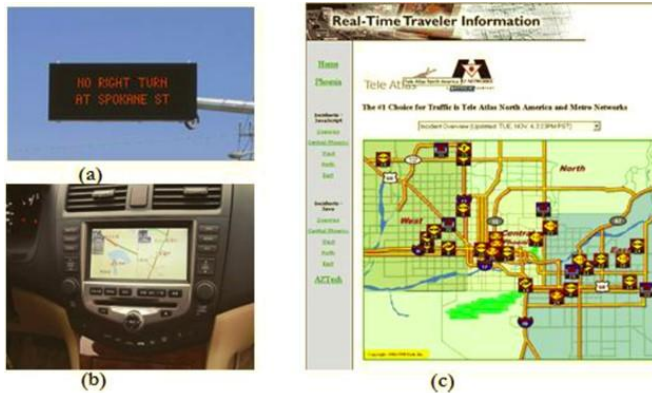


Gambar 2.7 Arsitektur Jaringan *OBU*

2.5 Variable messaging system (VMS)

Variable messaging system (sistem pesan berubah) dalam teknologi ITS, umumnya didefinisikan sebagai bagian dari *ATIS* (*Advanced Traveler Information System*), yang menyediakan informasi perjalanan dan lalu lintas yang diperlukan oleh pengendara/penumpang), seperti ditunjukkan pada gambar 2.8, *Variable messaging system* ini memberikan rute perjalanan bagi pengendara agar bisa mencapai tujuan yang diinginkan tanpa memakan waktu yang lama, karena diberikan arahan jalan yang minim dari kemacetan. *VMS* ini biasanya terdapat di

mobil atau pun bisa juga berada di *handphone*, contoh aplikasinya seperti Waze atau google maps.



Gambar 2.8 Penerapan VMS dalam Teknologi ATIS

Namun dalam penelitian ini VMS merupakan sebuah sistem yang kompleks, yang melakukan koordinasi dan manajemen pesan dari masukan sensor (posisi misalnya), keyboard (dari TMC misalnya) atau tombol (emergency) dan media input lainnya, yang ditampilkan dalam pesan dengan berbagai bentuk atau media berupa:

- Menu dan Display VMS untuk pengemudi di armada AMC
- Menu dan Display VMS untuk penumpang di armada AMC
- Menu dan Display VMS untuk calon penumpang di halte/depo AMC
- Display VMS untuk pengendara di atas badan jalan

Sehingga dari kompleksitas VMS tersebut dalam penelitian ini akan difokuskan penyediaan *fitur* Menu dan Display VMS untuk pengemudi dan penumpang di armada AMC dan calon penumpang di halte AMC.[1]

2.6 Quality of Service (QoS)

Quality of Service (QoS) didefinisikan sebagai deskripsi atau pengukuran kinerja keseluruhan suatu layanan, seperti jaringan telepon atau komputer terutama kinerja yang dilihat dan dirasakan oleh pengguna jaringan. Untuk mengukur kualitas layanan, beberapa aspek terkait dari layanan jaringan yang dipertimbangkan adalah packet loss, bit rate, *throughput*, *delay* transmisi, ketersediaan, jitter, dll.

Kualitas layanan diukur dengan standar dari *Telecommunication and Internet Protocol Harmonization Over Networks* (TIPHON) dan standar dari *International Telecommunication Union-Telecommunication* (ITU-T) *recommendation* Y1541. Dalam tugas akhir ini parameter QoS yang akan dianalisis adalah *throughput* dan *delay*.

2.6.1 Throughput

Throughput diartikan sebagai laju data aktual per satuan waktu (*bits per second*). Biasanya *throughput* selalu dikaitkan dengan *bandwidth*. Karena *throughput* memang bisa disebut sebagai *bandwidth* dalam kondisi yang sebenarnya. *Bandwidth* lebih bersifat tetap sementara *throughput* sifatnya dinamis tergantung *trafik* yang sedang terjadi. Cara untuk menghitung *throughput* sebagai berikut :

$$\text{throughput} = \frac{\text{jumlah data yang diterima}}{\text{waktu pengiriman}}$$

Dengan mendapat nilai *throughput* dari sebuah sistem yang sedang berjalan, kita bisa menentukan apakah sistem tersebut sudah memiliki kualitas yang baik atau belum. Apabila belum mendapatkan kualitas yang baik mungkin ada permasalahan dari sistem yang dibuat, atau pun bisa terjadi adanya permasalahan dalam jaringan yang dipakai. [10]

2.6.2 Delay

Salah satu parameter untuk menilai QOS (*Quality of Service*) dari sebuah jaringan adalah *delay*. *Delay* atau waktu paket di dalam sistem adalah waktu sejak paket tiba ke dalam sistem, sampai paket selesai ditransmisikan. Salah satu jenis *delay* adalah *delay* transmisi, yaitu waktu yang dibutuhkan untuk sebuah pengirim mengirimkan sebuah paket. *Delay* dapat dipengaruhi oleh kongesti, media fisik, jarak atau

juga waktu proses yang lama. Berikut adalah parameter kualitas sebuah jaringan dilihat dari besarnya *delay* menurut ITU-T.

Tabel 2.1 Standarisasi *Delay* ITU-T

Kategori Delay	Besar Delay
Excellent	< 150 ms
Good	150 – 300 ms
Poor	300 – 450 ms
Unnacceptable	>450 ms

Untuk menghitung *delay* dapat dihitung dengan:

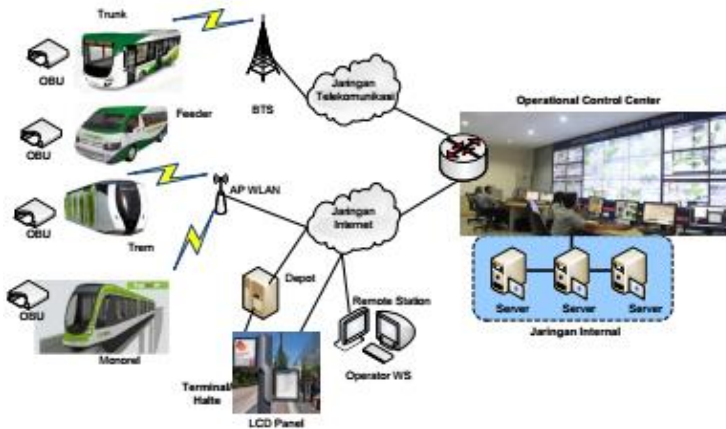
$$Delay = Waktu\ paket\ diterima - waktu\ paket\ dikirimkan$$

2.7 Angkutan Masa Cepat (AMC)

Dalam upaya mengatasi permasalahan kota Surabaya, diantaranya yaitu kemacetan, keterbatasan lahan untuk pembangunan jalan, perjalanan penumpang yang tidak efisien, dan meningkatnya polusi udara atas penggunaan kendaraan pribadi maka di butuhkan upaya penyediaan angkutan umum massal yang merupakan sarana transportasi berkelanjutan. Di sisi lain ketersediaan Angkutan Umum Massal Cepat (AMC) yang mampu menghubungkan prasarana transportasi utama di Kota Surabaya diharapkan mampu meningkatkan mobilitas orang secara efisien terhadap wilayah Surabaya dan sekitarnya sehingga dapat menarik investasi yang lebih besar [10]

2.7.1 Sistem Jaringan Komunikasi

Berangkat dari desain manajemen armada kota Surabaya dapat dibagi dan didefinisikan dari masing-masing bagian yang menjadi pekerjaan yang disusun sebagai tahap-tahap penerapan manajemen armada kota Surabaya. Berikut desain manajemen armada kota Surabaya:



Gambar 2.9 Desain Manajemen Armada AMC

Adapun tahap-tahap penerapan manajemen armada kota Surabaya sebagai berikut :

1. *Phase 1 CCROOM-Feeder Monitoring*

Pembangunan CCROOM yang disiapkan untuk mampu memonitoring *feeder vehicle* dan halte. Pembangunan ini membutuhkan pengadaan jaringan *hardware* dan *software* untuk internal CCROOM untuk meningkatkan kemampuan yang sekarang sudah ada.

2. *Phase 2 OBU Implementation*

Pada *phase* ini diawali dengan mengadakan *OBU* yang mempunyai modul GSM/GPS, *central unit*, piranti *human interface*, *diagnostic adapter*, modul I/O, *unit power supply* dan baterai yang kemudian diimplementasi pada masing-masing *unit* armada dan dilakukan uji coba monitoring.

3. *Phase 3 Fleet Optimation*

Melakukan optimasi armada dengan menggunakan *software* yang mampu memberikan rekomendasi *routing* yang efisien dan berskala bisnis sehingga bisa memberikan penghematan bahan bakar, mengurangi waktu tempuh, dan waktu kerja sehingga dapat meningkatkan *revenue* dan mengurangi biaya operasi

4. Phase 4 Operation and Maintenance

Phase ini untuk memberi jaminan bahwa sistem informasi dapat berfungsi sesuai dengan yang direncanakan dan memberikan unjuk kerja yang optimal sampai sistem habis masa penggunaannya.

BAB III

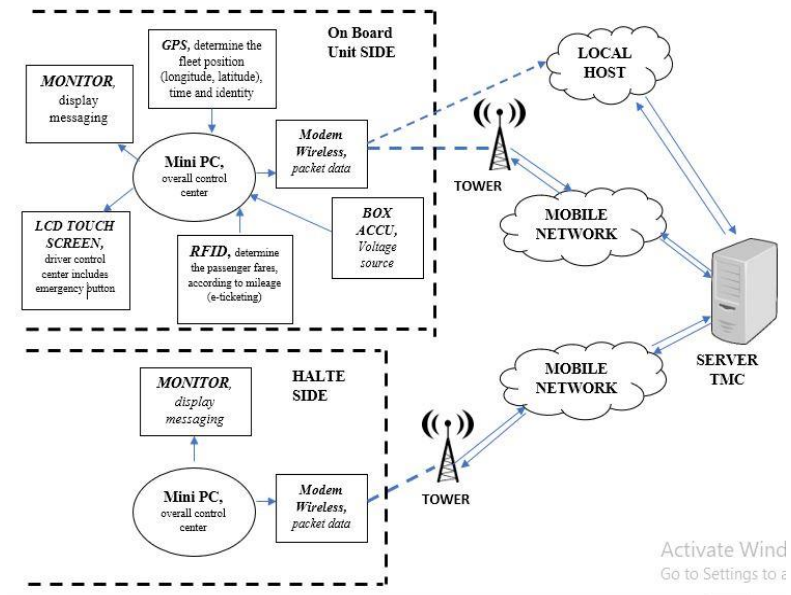
PERANCANGAN DAN IMPLEMENTASI SISTEM

Pada bab ini akan membahas tentang proses perancangan dan implemetasi komunikasi pengiriman data dari sejumlah titik *OBU* ke *Server* dengan menggunakan *Borland Delphi 6*. Untuk memudahkan proses perancangan dan implementasi sistem diperlukan alur yang menjelaskan garis besar proses yang dilakukan saat melaksanakan tugas akhir.

Selain perancangan pengiriman data, pada bab ini pun menjelaskan perancangan dari *fitur-fitur* yang ada di dalam aplikasi *OBU* yang telah dibuat sebelumnya serta media penyimpanan yang dibuat untuk menyimpan segala bentuk transaksi dan data yang telah dikirimkan.

3.1 Perancangan Sistem Komunikasi Terintegrasi

Sistem jaringan komunikasi yang akan diterapkan pada tugas akhir ini dapat dilihat pada arsitektur di bawah ini :



Gambar 3.1 Arsitektur Perpesanan

Pada gambar perancangan perpesana yang terintegrasi di atas terbagi menjadi dua *side* , pertama pada *On Board Unit Side* dan kedua pada *Halte Side*

Setiap *side* di sini didalamnya terdapat *On-Board Unit (OBU)* sebagai alat yang digunakan untuk pengiriman dan penerimaan informasi. Pada *OBU* dan Halte akan di *install* aplikasi yang telah dibuat menggunakan *software Borland Delphi 6*. Untuk mengirimkan data perpesanan menuju

Server (CCROOM), *Client* harus memasukkan beberapa data sesuai dengan kolom yang terdapat pada aplikasi. Kemudian data dikirim ke *Server*. Pengiriman data dari *OBU* ataupun dari Halte keduanya memakai jaringan seluler, di mana di tempatkan Sebuah alat Modem *Wireless* yang paket datanya menggunakan kartu *GSM 3*. Setelah data diterima oleh *Server*, *Server* akan memberi balasan ke *Client*. Selanjutnya data dimasukkan ke *database* untuk diolah. Data yang masuk ke *database* yaitu posisi setiap armada yang beroperasi (*longitude & latitude*), waktu pengiriman berupa tanggal, *time sent*, *time received*.

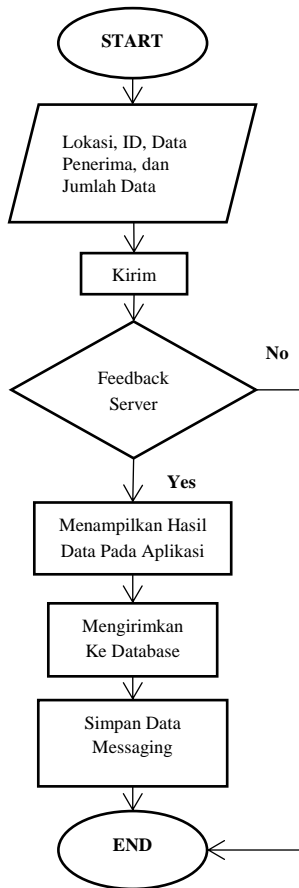
3.1.1 Perangkat *Server* yang Digunakan

Server yang digunakan untuk merecord data yang didapat adalah *aj01.lawanghosting.pw/cpanel*, di mana di dalamnya terdapat *database* keseluruhan yang menyimpan data-data yang didapatkan dari setiap armada yang beroperasi. Dari data yang didapat, *Server* akan otomatis mengirimkan kembali informasi yang didapat menuju halte ataupun armada untuk memberitahukan waktu tempuh serta tujuan selanjutnya bagi para penumpang ataupun calon penumpang yang ada di halte.

3.1.2 Perancangan Aplikasi *On Board Unit (OBU)*

Perancangan perangkat lunak *OBU (On Board Unit)* ini sebagai aplikasi untuk di *install* di *OBU (On Bard Unit)* yang ditempatkan pada setiap bus yang beroperasi untuk memonitoting posisi armada berada. Di mana saat armada bus bergerak maka aplikasi ini akan otomatis merekam perjalanan secara *real time*, serta mengirimkan posisi dan waktu tempuh kepada *Server CCROOM (Control Center Room)*.

Berikut adalah perancangan aplikasi pengiriman data yang dijalankan *OBU*:



Gambar 3.2 *Flowchart Aplikasi OBU*

Pengiriman posisi di sini terecord dalam bentuk data latitude dan longitude yang di kirimkan menuju *Server CCROOM (Control Center Room)*, kemudian pada *Server CCROOM (Control Center Room)* akan membalas pesan yang dikirimkan oleh armada dan diteruskan kembali menuju *OBU (On Board Unit)* di tiap armada agar perpesanan tersebut dapat disampaikan kepada para penumpang untuk pemberhentian selanjutnya.

Pesan balasan yang dikirimkan oleh *Server* akan ditampilkan di monitor yang ditempatkan di setiap armada, dengan tujuan agar para penumpang dapat mengetahui pemberhentian selanjutnya di halte mana dan memerlukan waktu berapa lama untuk mencapai tujuan yang diinginkan. Berikut tampilan dari aplikasi yang telah dirancang untuk pengiriman pesan:



Gambar 3.3 Tampilan Aplikasi Delphi 6

Berikut Penjelasan setiap button yang ada di aplikasi Delphi 6:

1. Koneksi komponen GPS pada port berapa
2. Panel indicator bahwa data yang dikirimkan masuk atau tidak, apabila indicator angkanya sama maka data tidak masuk, begitu pula kebalikannya
3. Menunjukan tanggal serta penunjuk waktu
4. Untuk menunjukkan hasil data yang sudah terseleksi (latitude & longitude)
5. Menentukan identitas kendaraan berdasarkan plat nomer yang akan dikirimkan menuju *Server*
6. Menampilkan link yang akan dituju untuk selanjutnya ditampilkan hasil posisi pada web
7. Hasil data keseluruhan yang dikirimkan oleh armada

3.1.3 Perancangan *Database* dan *Web Report* untuk Penyimpanan Data

Server yang digunakan untuk merecord data yang didapat adalah *aj01.lawanghosting.pw* dengan memanfaatkan *database PHPMYadmin* pada *Server*. Data yang dikirimkan setiap armada masuk ke *Server* akan disimpan didalam *database data1pos*. berikut penjelasan masing-masing kolom yang berada di *database*:



Table	Action	Rows	Type	Collation	Size	Overhead
absen_driver	Browse Structure Search Insert Empty Drop	8	MyISAM	latin1_swedish_ci	2.6 KiB	292 B
data1	Browse Structure Search Insert Empty Drop	1,637	MyISAM	latin1_swedish_ci	75 KiB	-
data1pos	Browse Structure Search Insert Empty Drop	11	MyISAM	latin1_swedish_ci	1.6 KiB	-
emergency	Browse Structure Search Insert Empty Drop	22	MyISAM	latin1_swedish_ci	4.1 KiB	788 B
JadwalBus	Browse Structure Search Insert Empty Drop	3	MyISAM	latin1_swedish_ci	2.2 KiB	-
user_login	Browse Structure Search Insert Empty Drop	2	MyISAM	latin1_swedish_ci	1 KiB	-
6 tables	Sum	1,683	MyISAM	latin1_swedish_ci	86.5 KiB	1 KiB

Gambar 3.4 Tampilan Tabel *Database* di *Server*

Pada gambar di atas menunjukkan tabel yang berada di *database Server*. Dalam tabel tersebut ada beberapa field yaitu:

1. *absen_driver*, berisi *database* absensi *driver* sebelum mengoperasikan kendaraannya.
2. *data1*, berisikan data-data *OBU* yang dikirimkan oleh armada yang beroperasi.
3. *data1pos*, hampir sama dengan *data1* berisikan data-data *OBU* yang dikirimkan armada yang beroperasi berupa lokasi, waktu, identitas armada.
4. *emergency*, berisikan data yang dikirimkan *OBU* apabila armada dalam keadaan darurat seperti ban bocor, penumpang sakit, kecelakaan
5. *JadwalBus*, berisikan jadwal keberangkatan dan kedatangan bus yang ditempatkan di halte
6. *user_login*, berisikan *username* sekaligus *password*

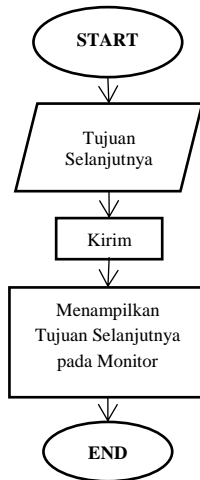
Selain *database* yang sudah dirancang, media penyimpanan lainnya adalah pada web report, seluruh aktivitas yang berkaitan dengan OBU keseluruhan akan ditempatkan di dalamnya, mencakup absensi *driver*, pemantauan armada, *emergency button* dan lain lainnya. Semua bisa dicek di aj01.lawanghosting.pw/report.



Gambar 3.5 Tampilan *Web Report*

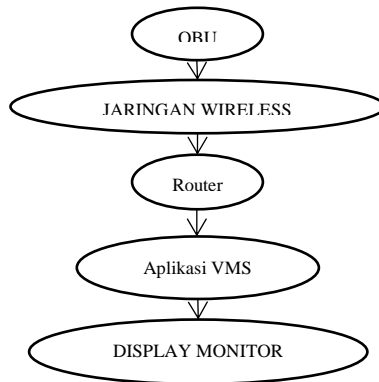
3.2 Perancangan Perpesanan Penumpang

Pada bagian ini akan di jabarkan mengenai sistem perpesanan penumpang, di mana perpesanan penumpang ini akan membantu penumpang yang berada di dalam bus untuk mengetahui pemberhentian selanjutnya, kemudian dapat memprediksi waktu tempuh dari halte 1 menuju halte selanjutnya. Di setiap armada akan ditempatkan sebuah monitor yang berfungsi menampilkan informasi tujuan selanjutnya agar lebih mempermudah para penumpang mengetahui halte mana yang akan di tuju oleh armada yang ditumpangi.



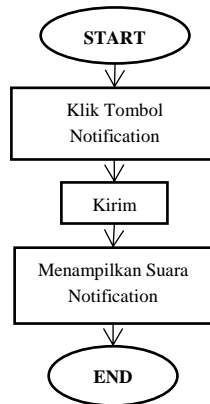
Gambar 3.6 *Flowchart* Perpesanan Penumpang

Di dalam bus, terdapat LCD *Touchscreen* yang berfungsi untuk memberi informasi kepada penumpang halte selanjutnya yang akan di tuju, saat *fitur* di LCD *Touchscreen* di tekan oleh *driver*, maka tujuan selanjutnya akan ditampilkan pada monitor yang ditempatkan pada bagian depan bus agar seluruh penumpang mengetahui kemana tujuan selanjutnya dan membutuhkan waktu berapa lama.



Gambar 3.7 *Flowchart* Informasi VMS Penumpang

Saat *driver* meng *klik* aplikasi OBU yang di installkan di dalam *mini pc* untuk halte selanjutnya pada *LCD Touchscreen* yang ditempatkan di samping kemudi, informasi tujuan selanjutnya akan dialirkan menuju monitor yang ditempatkan di dalam armada, dengan melewati jaringan wireless yang paket datanya di alirkan menuju router kemudian menuju ke aplikasi VMS yang di tempatkan di dalam armada dan di tampilkan di monitor yang ditempatkan di dalam armada yang beroperasi.

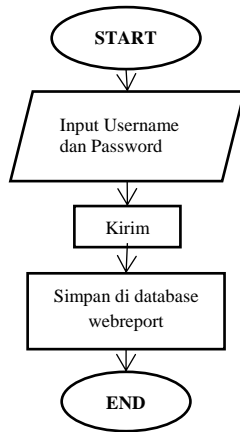


Gambar 3.8 *Flowchart Button Notification*

Selain itu terdapat sebuah tombol untuk para penumpang agar mempermudah untuk memberikan kode bahwa penumpang akan turun di halte terdekat.

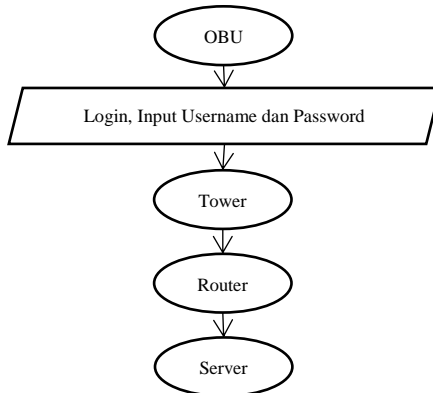
3.3 Perancangan Perpesanan *Driver*

Pada bagian ini akan dijabarkan mengenai sistem perpesanan *driver*, di mana perpesanan *driver* ini akan membantu *driver* dalam memberikan informasi kepada seluruh penumpang melalui sebuah alat yang ditempatkan di dekat kemudi. Alat tersebut berupa *LCD Touchscreen* yang didalamnya sudah terinstall aplikasi pendukung dengan berbagai fitur didalamnya. Fitur-fitur tersebut diantaranya absensi *driver*, *emergency button*, VMS penumpang.



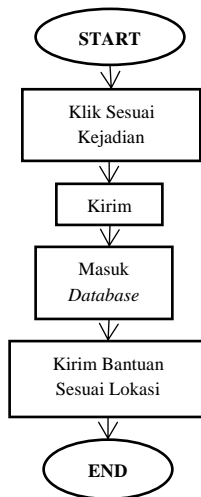
Gambar 3.9 *Flowchart Absensi Driver*

Pada saat *driver* ingin mengoperasikan armadanya, para *driver* diwajibkan untuk absen terlebih dahulu, agar manajemen dapat merecord apakah *driver* tersebut datang sesuai dengan peraturan atau tidak. *Driver* akan mendapatkan *username* dan *Password* untuk login kedalam sistem yang akan digunakan untuk mengoperasikan kendaraannya, setelah login maka otomatis terecord oleh *Server* dan di simpan di *database* dan *web report*.



Gambar 3.10 *Aliran Informasi Absensi Driver*

Paket data dikirimkan menuju *server* berawal dari *login* aplikasi yang akan digunakan sebagai media informasi, kemudian paketan tersebut dikirim dengan rute ke *tower* terdekat dengan tempat pengiriman, kemudian paket data tersebut akan di alirkan menuju *router* dari *provider* yang digunakan yaitu TRI dan selanjutnya dari *router* tersebut otomatis mengirimkan informasi absensi ke *server*, untuk selanjutnya sebagai pendataan pihak administrasi *server* berada di alamat url *aj01.lawanghosting.pw*

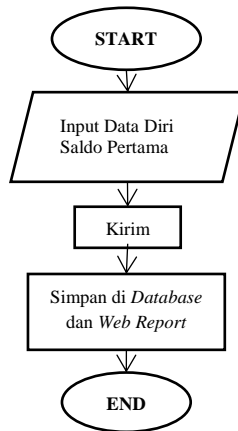


Gambar 3.11 *Flowchart Emergency Button*

Selain absensi *driver*, pada perpesanan *driver* memiliki fitur lain yaitu *emergency button*, yang berfungsi untuk memberikan informasi dari armada menuju *server* secara *real time* apabila sedang mengalami sebuah kejadian yang tidak diinginkan seperti bocor ban, penumpang sakit, kecelakaan, dan mogok. Pengoperasian dari *emergency button* ini sama pada halnya sistem pengiriman informasi pada pemantauan armada, pada aplikasi OBU tersebut *driver* hanya meng klik sesuai kejadian yang terjadi, kemudian setelah di klik maka pesan akan dikirimkan menuju *server*, dan tindakan selanjutnya yaitu pusat akan mengirimkan bantuan sesuai kejadian dan lokasi yang di berikan oleh armada.

3.4 Perancangan *Fitur Ticketing*

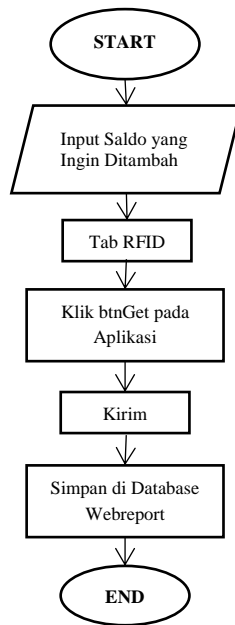
Pada perancangan *fitur* pada *ticketing* ini diharapkan sistem kerja *ticketing* dapat seperti *fitur ticketing* pada umumnya, dimana sebuah RFID berfungsi sebagai sebuah alat untuk melakukan transaksi atau sebagai alat pembayaran dengan mempunyai tarif yang berbeda-beda bagi setiap profesi. Tarif sudah di tentukan yaitu untuk pelajar/mahasiswa Rp. 3000, untuk dosen Rp. 2000, sedangkan untuk umum yaitu Rp. 4000.



Gambar 3.12 *Flowchart* Registrasi Member Baru

Untuk mengaktifkan RFID yang akan dipakai sebagai alat pembayaran perjalanan bagi para calon penumpang, hal yang harus diperhatikan adalah dengan meregistrasi terlebih dahulu RFIDnya, dengan memasukkan data diri terlebih dahulu seperti nama, NIK, dan memasukkan saldo pertama yang akan di isi. Setelah registrasi telah dilakukan maka RFID sudah siap di gunakan untuk melakukan perjalanan.

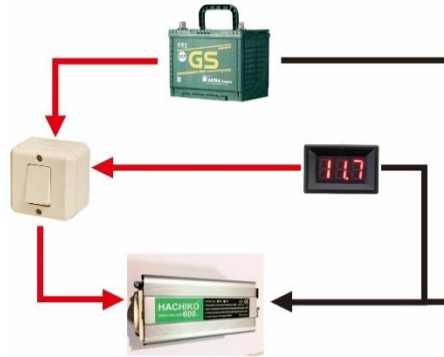
Setelah kartu sudah di pakai berkali-kali pastinya saldo semakin lama semakin habis dan menipis, maka dari itu harus diisi kembali dengan men *top up* saldo yang dimiliki, dengan pada aplikasi hanya menambahkan berapa saldo yang diinginkan kemudian tapkan RFIDnya setelah itu klik btinget dan otomatis e-ticketing telah bertambah saldonya.



Gambar 3.13 *Flowchart* Top up Saldo

3.5 Perancangan Sumber Tenaga pada Armada

Dengan pengiriman data yang terus berjalan secara *real time*, berjalan dengan terus menerus pasti membutuhkan sumber daya yang mendukung alat serta sistem yang berjalan, perancangan sumber daya ini bertujuan agar tenaga yang di berikan kepada alat dan armada yang beroperasi tidak kurang dan selalu cukup dalam proses pengiriman serta penerimaan informasi. Berikut rangkaian sumber tenaga yang akan dipasangkan di tiap armada:



Gambar 3.14 Rangkaian Box Aki

Rangkaian aki ini berfungsi sebagai wadah dari aki kering 70ah, didalam rangkaian memiliki inverter yaitu pengubah tegangan DC menjadi tegangan AC. Rangkaian Box aki ini dapat dimatikan dengan cara menekan tombol saklar maka tidak ada sumber tegangan yang mengalir ke *inverter* dan voltmeter digital. Untuk *charging* aki dapat dilakukan dengan cara melepas skun kabel yang terpasang pada inverter lalu dipasang ke charger aki.

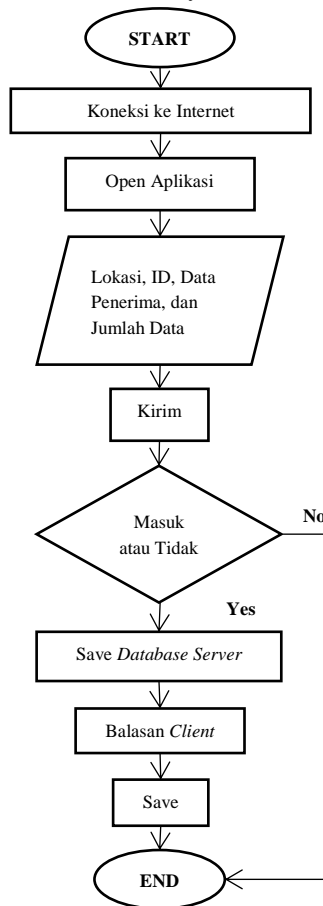
3.6 Implementasi Sistem Komunikasi Terintegrasi

Pada pengerjaan tugas akhir ini, sistem dan aplikasi yang telah dibuat akan diimplementasikan. Implementasi yang dimaksud di sini adalah pengiriman informasi dari *Client* (armada yang beroperasi) menuju *Server* ataupun sebaliknya. Selain pengiriman informasi dari *Client-Server* implementasi sistem yang di buat selalu menggunakan jaringan wireless.

3.6.1 Implementasi Sistem

Dalam pengerjaan tugas akhir ini, implementasi sistem ini terhubung dengan *server* `aj01.lawanghosting.pw/cpanel` terdapat pada PHPMyadmin yang didalamnya memiliki *database* keseluruhan yang menyimpan data-data yang didapatkan dari setiap armada yang beroperasi. Kemudian Aplikasi *OBU* yang digunakan oleh *client* untuk mengirim data telah dibuat dengan menggunakan *software* Borland Delphi 6.

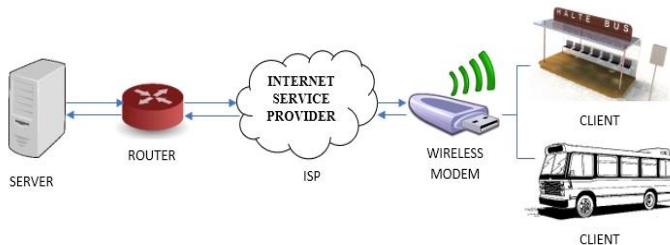
Yang harus dilakukan pertama kali adalah yang pasti *client* wajib terkoneksi dengan jaringan internet. Setelah itu, *client* akan memasukkan data yang akan di kirimkan menuju ke *server*, berupa ID yang berfungsi untuk mengetahui identitas armada yang beroperasi, setelah itu aplikasi akan otomatis mendetect lokasi dimana armada itu berada kemudian data lokasi yang telah di dapat akan disimpan di dalam *database* yang sudah dibuat sebelumnya.



Gambar 3.15 Flowchart Implementasi Sistem

3.6.2 Implementasi Jaringan

Jaringan yang dipakai pada tugas akhir yang dikerjakan ini menggunakan jaringan *wireless* yang di tempatkan di setiap armada ataupun halte. Terdapat sebuah modem *wireless* dengan operator 3 (three) yang memiliki paket data untuk menghubungkan antara *client* menuju *server*.



Gambar 3.16 Implementasi Sistem Jaringan

Pertama yang dilakukan adalah koneksikan sistem dan aplikasi yang dibuat ke internet melalui modem *wireless*, setelah itu apabila ingin mengirimkan informasi dari *client* menuju *server* dengan membuka aplikasi yang sudah di *install* lkan di dalam armada masukkan identitas armada lalu *klik button* maka akan terlihat posisi armada tersebut berada dikoordinat berapa, yang kemudian akan otomatis terkirim menuju *server* dan data yang dikirimkan akan disimpan di dalam *database* yang telah di buat sebelumnya.

Selain menggunakan jaringan *wireless*, sistem ini juga menggunakan jaringan *local host*, fungsinya apabila saat pengiriman data mengalami gangguan koneksi maka data-data yang dikirimkan akan otomatis tersimpan juga di dalam jaringan *local host* yang dibuat, bisa dibilang sebagai media penyimpanan cadangan agar data-data yang dikirimkan tetap tersimpan dengan baik tanpa ada kekurangan data sedikitpun.

3.7 Skenario Pengujian Sistem

Setelah proses implementasi sistem telah selesai, langkah selanjutnya adalah menentukan skenario pengujian sistem, apa saja yang akan di uji dan apakah pengujian tersebut sudah sesuai dengan implementasi yang dijelaskan sebelumnya atau belum. Pengujian sistem yang di rancang meliputi pengujian sistem *client-server* serta *fitur* lainnya yang ada di dalam OBU dan pengujian jaringan. Pada pengujian jaringan, dibuat skenario pengujian *Quality of Service* dari banyaknya parameter yang

berkaitan dengan *QOS* dalam tugas akhir ini hanya mencari nilai *throughput* dan *delay* nya saja.

Pengujian setiap *fitur* yang ada pada OBU memiliki target luaran yang beragam sesuai dengan fungsi yang diterapkan dari masing-masing *fitur* yang ada. Berikut adalah syarat-syarat dari perancangan sistem yang telah dirancang sebelumnya:

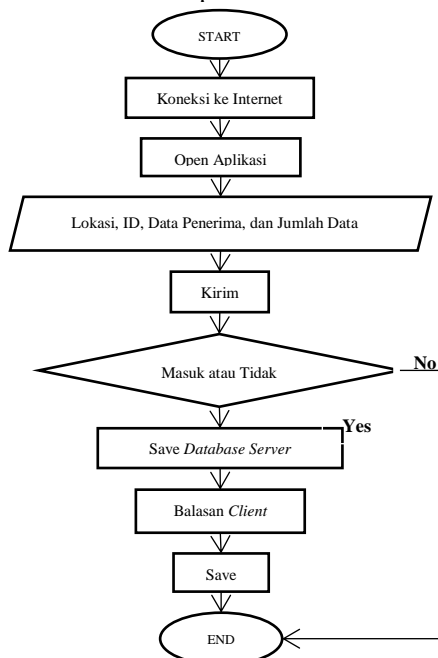
Tabel 3.1 Syarat Perancangan Sistem

<i>Fitur-Fitur</i>	Target Perancangan
Pemantauan armada	Dapat melakukan pengiriman Informasi berupa waktu, posisi, dan identitas dari armada menuju <i>server</i> pada gambar 3.15
Perpesanan Penumpang (VMS Penumpang)	Dapat menampilkan halte selanjutnya (terdekat) di monitor yang ditempatkan dalam armada Pada gambar 3.7
<i>Button Notification</i>	<i>Button notification</i> , dapat berfungsi (mengeluarkan suara) dan memberikan informasi kepada <i>driver</i> pada gambar 3.8
<i>Emergency Button</i>	Dapat mengirimkan informasi ke <i>server</i> , lokasi armada berada, dan <i>server</i> akan mengirimkan bantuan sesuai lokasi dan kejadian darurat yang di laporkan. Pada gambar 3.11
Absensi <i>Driver</i>	<i>Input</i> data <i>login</i> yaitu waktu, data diri, dan tanggal dapat terrekam didalam <i>database</i> dan <i>web report</i> pada gambar 3.9
<i>E-Ticketing</i>	Registrasi member baru, tambah saldo dan tapin RFID dapat berfungsi dan terrekam ke dalam <i>web report</i> 3.12 dan 3.1
Pengujian <i>Throughput</i>	Mengetahui nilai <i>Throughput</i> maksimal, minimal serta nilai rata-rata proses pengiriman informasi dari OBU menuju <i>server</i>

Pengujian <i>Delay</i>	Mengetahui nilai <i>Delay</i> maksimal, minimal serta rata-rata proses pengiriman informasi dari OBU menuju <i>server</i> pada gambar 3.19
------------------------	--

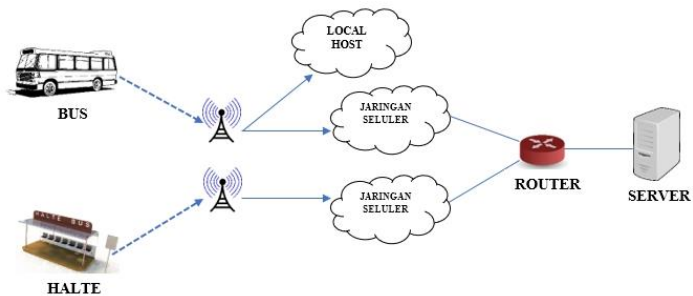
3.7.1 Skenario Pengujian Sistem

Pada bagian ini akan dijabarkan mengenai skenario pengujian sistem *client-server* komunikasi terintegrasi Angkutan Masal Cepat (AMC) yang telah dibuat pada bagian halte *side* maupun di bagian *OBU side*. Pengujian ini bertujuan untuk mengetahui apakah sistem telah sesuai dengan rancangan atau belum. Pengujian ini dilakukan dengan cara melakukan proses pengiriman data dari *client* ke *server* menggunakan *software* Borland Delphi yang telah dibuat dan di *install* pada perangkat *OBU* di setiap armada komunikasi terintegrasi AMC. Proses ini dilakukan untuk mengetahui apakah data yang dikirimkan dapat diterima oleh *server* dan *client* mendapat balasan dari *server*.



Gambar 3.17 Flowchart Pengujian *Client-server*

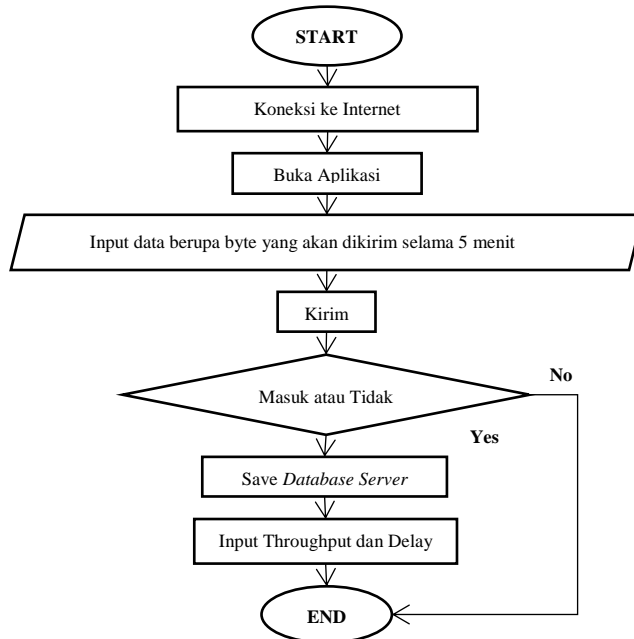
Selain itu, pengujian ini juga dilakukan untuk mengetahui apakah data yang diterima oleh *server* dapat disimpan dalam *database server* atau tidak, apakah perpesanan *driver* dapat di tampilkan di monitor bus atau tidak, serta perpesanan dapat dilakukan dengan baik di setiap *side* nya. Dan setelah pengiriman berhasil, dan pada gambar di bawah ini di tunjukkan proses informasi yang di kirimkan dari armada menuju *server*. Pengiriman informasi yang di kirimkan berupa titik kordinat (latitude dan longitude), waktu, serta identitas armada melalui jaringan komunikasi wireless yang di tempatkan di setiap armada menggunakan paket data provider 3 (tri), yang di teruskan melalui *router* sebelum sampai ke database yang di tempatkan di *server aj01.lawanghosting.pw*.



Gambar 3.18 Proses Informasi dari Armada ke Server

3.7.2 Skenario Pengujian Jaringan

Berikut adalah *flowchart* skenario pengujian jaringan yang akan di lakukan pada tugas akhir yang di kerjakan ini:



Gambar 3.19 *Flowchart* Pengujian Jaringan

Pada skenario jaringan ini, akan dilihat dan di amati QoS jaringan yang telah di rancang sebelumnya. QoS yang di tinjau adalah berapa nilai Throughput serta seberapa besar *Delay* yang di dapat. Langkah-langkah pengujian yang dilakukan yaitu aplikasi telah terkoneksi dengan jaringan Wi-Fi atau jaringan salah satu operator seluler yaitu operator three (3).

[Halaman ini sengaja dikosongkan]

BAB IV

PENGUJIAN DAN ANALISIS SISTEM

Pada bab ini akan di jelaskan hasil pengujian dan analisis dari perancangan sistem komunikasi terintegrasi yang sudah dijelaskan pada bab sebelumnya, serta menjelaskan beberapa *fitur* lainnya yang ada di dalam OBU. sistem ini menggunakan aplikasi yang di buat dengan *Borland Delphi 6* dimulai dari proses pengiriman informasi dari oleh *client* hingga dapat diterima oleh *server*. Pengujian ini di bagi menjadi dua bagian ya itu pada bagian *OBU side* yaitu di tiap armada apakah aplikasi *client-server* ini dapat mengirimkan informasi, memanfaatkan *fitur* yang ada sesuai dengan implementasi atau tidak dan yang kedua pada bagian halte *side* apakah dapat menampilkan informasi pada monitor yang ditempatkan di halte atau tidak, serta menganalisa kualitas data yang dikirimkan apakah sesuai standar atau tidak.

4.1 Pengujian Sistem pada *OBU side*

Sesuai dengan skenario yang telah di jelaskan pada bab 3 sebelumnya, pada bab ini akan dilakukan pengujian sistem *client-server* yang di buat sebelumnya. Pengujian ini bertujuan untuk mengetahui apakah informasi yang dikirimkan dari armada dapat terkirimkan ke *server* dan dapat di simpan di dalam *database* atau tidak, ataupun sebaliknya informasi yang telah di terima *server* apakah bisa melakukan balasan kembali menuju armada yang sedang beroperasi atau tidak.

Sebelum menguji Sistem yang telah di rancang, akan dilakukan pengujian modul GPS Neo Starter Kit dengan menggunakan GPS HP. Pengujian ini dilakukan untuk mengetahui keakuratan dari modul GPS dengan membandingkan GPS yang ada di HP. Pengujian ini dilakukan dengan cara mengambil titik koordinat latitude dan longitude suatu tempat dengan jarak terpisah 100M. Modul GPS tidak dapat mendeteksi titik koordinat ketika memakai sumber dari laptop. Ketika memakai power bank titik koordinat dapat muncul. Berikut adalah tabel hasil pengukurannya:

Tabel 4.1 Hasil Pengukuran Modul GPS

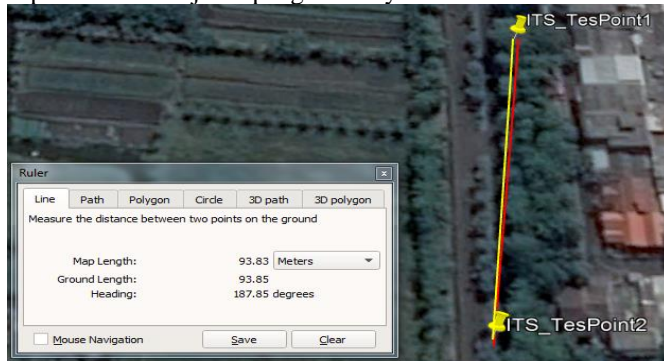
No	Jarak	Modul GPS		GPS HP	
		Latitude	Longitude	Latitude	Longitude
1	100m	7.17.1	112.47.58	7.17.2	112.47.58
2	100m	7.17.4	112.47.57	7.17.4	112.47.57
3	100m	7.17.6	112.47.57	7.17.7	112.47.57
4	100m	7.17.8	112.47.54	7.17.8	112.47.54
5	100m	7.17.8	112.47.51	7.17.9	112.47.51
6	100m	7.17.9	112.47.48	7.17.9	112.47.48
7	100m	7.17.9	112.47.45	7.17.9	112.47.45
8	100m	7.17.10	112.47.41	7.17.10	112.47.41
9	100m	7.17.10	112.47.38	7.17.10	112.47.38
10	100m	7.17.9	112.47.36	7.17.9	112.47.36
11	100m	7.17.6	112.47.36	7.17.6	112.47.36
12	100m	7.17.3	112.47.35	7.17.3	112.47.35
13	100m	7.16.59	112.47.35	7.16.59	112.47.35
14	100m	7.16.57	112.47.36	7.16.57	112.47.36
15	100m	7.16.55	112.47.38	7.16.55	112.47.38
16	100m	7.16.54	112.47.32	7.16.53	112.47.32
17	100m	7.16.51	112.47.31	7.16.51	112.47.31
18	100m	7.16.48	112.47.28	7.16.48	112.47.28
19	100m	7.16.46	112.47.26	7.16.46	112.47.26
20	100m	7.16.44	112.47.24	7.16.44	112.47.24

Setelah didapatkan titik koordinat *latitude* dan *longitude* dimasukkan didalam *software Google Earth* untuk melihat apakah titik koordinat sesuai dengan tempat mengambil titik koordinat. Berikut adalah *point-point* dari titik koordinat.



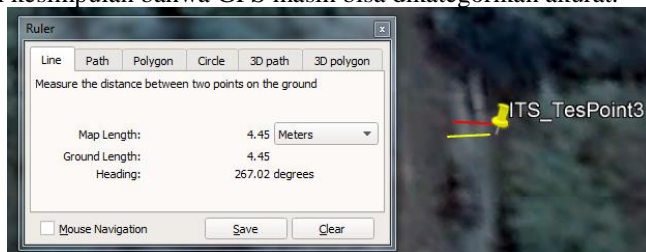
Gambar 4.1 Point to Point dari Titik Koordinat

Dari Google Earth didapatkan jarak antar point yaitu sebesar 93,83 M. Terdapat selisih dari jarak pengukuran yaitu sebesar 100M.



Gambar 4.2 Pengukuran Jarak Tespoint 1 dengan Lainnya

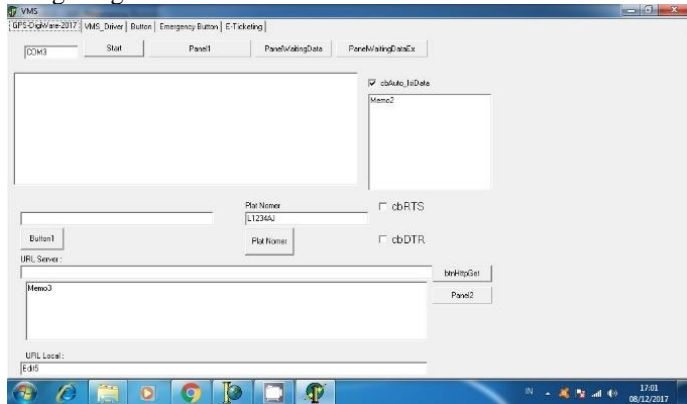
Jarak Point dengan tempat sebenarnya adalah 4,45 M. Jadi dapat diambil kesimpulan bahwa GPS masih bisa dikategorikan akurat.



Gambar 4.3 Jarak Tespoint

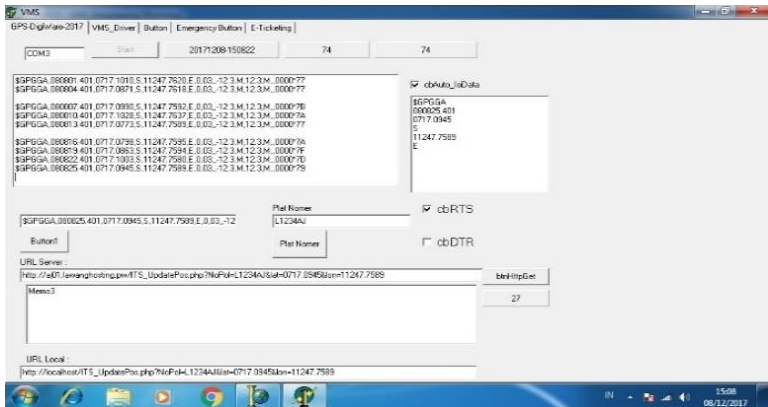
4.1.1 Pengujian *OBU* untuk Pemantauan Armada

Yang pertama harus dilakukan adalah koneksi ke internet, lalu pastikan semua *port* sudah terpasang dengan baik, lalu cek port gps apakah sudah tersambung atau belum, setelah itu pastikan panel indikator memiliki angka yang berbeda, karena apabila panel indikator menunjukkan angka yang sama maka tandanya alat tersebut belum tersambung dengan baik.



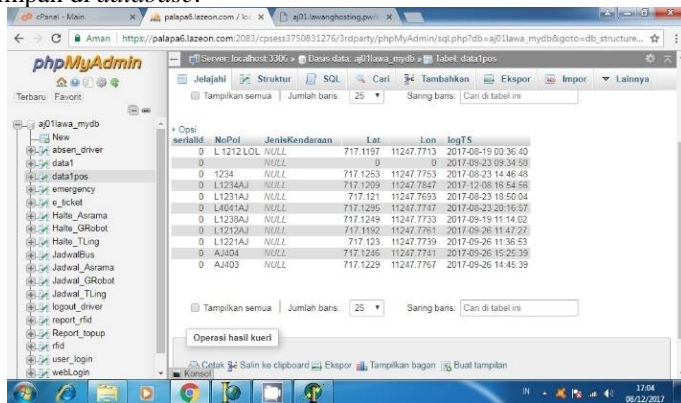
Gambar 4.4 Tampilan Awal Aplikasi

Setelah dipastikan semua sudah terkoneksi dengan baik, maka langkah selanjutnya adalah menginputkan identitas nomor polisi dari armada yang akan di tracking keberadaannya, lalu klik *start* maka aplikasi akan mengirimkan data posisi dari armada, data yang dikirim tersebut adalah *latitude* dan *longitude*, waktu pengiriman berupa tanggal, *time sent*, *time received*.



Gambar 4.5 Tampilan Hasil *Messaging*

Setelah di *input* dan di *start* maka akan muncul informasi dari armada yang sedang beroperasi dan data tersebut otomatis akan tersimpan di *database*.



Gambar 4.6 Hasil *Messaging* disimpan dalam *Database*

Selain tersimpan di dalam *database* data koordinat yang dikirimkan oleh armada ke *server* akan otomatis di simpan juga di dalam web report sebagai media penyimpanan yang sudah dibuat sebelumnya

Print

No	NOMER POLISI	LATITUDE	LONGITUDE	TIME
1	L1234AJ	7.28492	112.79603833333334	2017-12-08 15:08:58
2	L1234AJ	7.284423333333333	112.79602166666668	2017-12-08 15:10:35
3	L1234AJ	7.285905000000001	112.79634166666668	2017-12-08 15:12:38
4	L1234AJ	7.285286666666667	112.79620500000001	2017-12-08 15:14:33
5	L1234AJ	0	0	2017-12-08 15:15:59
6	L1234AJ	0	0	2017-12-08 15:18:10
7	L1234AJ	0	0	2017-12-08 15:21:50
8	L1234AJ	7.285404999999999	112.79623166666667	2017-12-08 15:32:23
9	L1234AJ	7.285413333333335	112.79620666666666	2017-12-08 15:33:46
10	L1234AJ	7.285406666666668	112.796265	2017-12-08 15:36:10
11	L1234AJ	7.28538	112.79619833333332	2017-12-08 15:38:19
12	L1234AJ	7.285386666666667	112.79622	2017-12-08 15:40:17
13	L1234AJ	7.285385000000001	112.79624666666665	2017-12-08 15:42:53
14	L1234AJ	7.2854	112.79625	2017-12-08 15:45:54
15	L1234AJ	7.285463333333333	112.79624666666665	2017-12-08 15:48:24
16	L1234AJ	7.285318333333334	112.79621499999999	2017-12-08 16:54:15

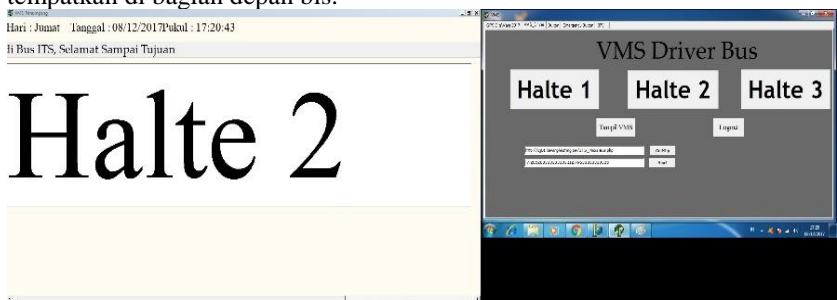
Dicetak Pada Tanggal : 08-12-2017

Gambar 4.7 Hasil *Messaging* pada *Web Report*

4.1.2 Pengujian *VMS* Penumpang

Pada *VMS* penumpang ini dibuat agar para penumpang yang berada di dalam armada mendapatkan kenyamanan serta kemudahan dalam memperoleh informasi pemberhentian mana selanjutnya yang akan ditampilkan pada monitor yang di tempatkan di dalam armada.

Pada dasarnya *OBU* yang ditempatkan dekat dengan kemudi, terdapat sebuah *fitur* dimana *driver* dapat memberikan informasi tujuan selanjutnya kepada para penumpang yang berada di dalam bis, dengan hanya mengetuk tujuan selanjutnya pada *LCD Touchscreen* yang berada dekat kemudi, setelah itu akan di tampilkan pada monitor yang di tempatkan di bagian depan bis.



Gambar 4.8 Tampilan Awal Aplikasi & Hasil pada Monitor

Selain untuk mengetahui tujuan selanjutnya, penumpang yang berada di dalam bis pun dapat memberikan kode kepada *driver* dengan menekan tombol untuk turun pada halte terdekat.

4.1.3 Pengujian VMS Driver

Sesuai dengan perancangan pada bab 3, pengujian *VMS driver* ini pertama yang harus dilakukan adalah sama seperti sebelumnya connect dengan aplikasi terlebih dahulu, kemudian aktifkan aplikasi perpesanan *driver*. Perpesanan *driver* disini meliputi absensi *driver* serta *emergency button* apabila armada mengalami sebuah kejadian yang tidak diinginkan.

Untuk absensi *driver* sendiri *driver* mendapatkan sebuah username serta *password* untuk *login* menuju aplikasi yang akan digunakan untuk mengoperasikan armadanya.



Gambar 4.9 Tampilan Absen Driver Sekaligus Login Aplikasi

Setelah masuk ke aplikasi dan klik tombol *login*, kemudian setelah *driver login* maka data diri *driver* akan langsung masuk ke dalam *database* serta *webreport* yang telah di rancang sebelumnya, data yang disimpan yaitu nama, *ID Driver*, dan waktu saat *login*.

	id	nama	xtag	time
	56	SamuelWicaksono	780027BBAC48	2017-11-27 14:44:03
	57	Edi2	EdiRfd	2017-11-27 14:45:08
	58	SlametiHaryadi	6F0085BFFFAA	2017-11-27 14:45:32
	59	SlametiHaryadi	6F0085BFFFAA	2017-11-27 14:45:32
	60	SlametiHaryadi	6F0085BFFFAA	2017-11-27 14:45:35
	61	SlametiHaryadi	6F0085BFFFAA	2017-11-27 14:45:35
	62	AdiSupono	7800278E4293	2017-11-27 14:46:01
	63	AdiSupono	7800278E4293	2017-11-27 17:47:40
	64	SlametiHaryadi	6F0085BFFFAA	2017-11-27 17:47:51
	65	AdiSupono	7800278E4293	2017-12-07 14:26:21
	66	AdiSupono	7800278E4293	2017-12-07 14:26:33

Gambar 4.10 Hasil Absensi di Database

21	SamuelWicaksono	2017-12-07 14:38:57
22	AdiSupono	2017-12-08 18:03:05
23		2017-12-09 19:02:08
24		2017-12-09 19:10:02
25		2017-12-09 19:10:13
26		2017-12-09 19:10:39
27		2017-12-09 19:13:22
28		2017-12-09 19:13:41
29		2017-12-09 19:13:45
30	AdiSupono	2017-12-09 20:32:50
31	AdiSupono	2017-12-13 14:05:17
32	AdiSupono	2017-12-13 15:22:49
33	AdiSupono	2017-12-13 22:25:28
34	AdiSupono	2017-12-14 09:06:11
35	AdiSupono	2017-12-14 09:16:35
36	AdiSupono	2017-12-14 09:18:35
37	AdiSupono	2017-12-14 09:33:54
38	AdiSupono	2017-12-14 09:42:42
39	AdiSupono	2017-12-14 09:48:04
40	AdiSupono	2017-12-14 09:58:46
41	AdiSupono	2017-12-14 10:07:50
42	AdiSupono	2017-12-14 10:31:43
43	AdiSupono	2017-12-14 10:38:22
44	AdiSupono	2017-12-14 10:47:24
45	AdiSupono	2017-12-14 10:49:41
46	AdiSupono	2017-12-14 10:52:03
47	AdiSupono	2017-12-14 10:53:14
48	AdiSupono	2017-12-14 15:18:17
49	AdiSupono	2017-12-14 15:47:02
50	AdiSupono	2017-12-14 15:49:33
51	AdiSupono	2017-12-14 16:07:01
52	AdiSupono	2017-12-14 16:11:28
53	AdiSupono	2017-12-14 16:13:08
54	AdiSupono	2017-12-14 16:26:03
55	AdiSupono	2017-12-14 16:32:30

Dicetak Pada Tanggal : 2017-12-07 14:26:21

Gambar 4.11 Hasil pada Web Report

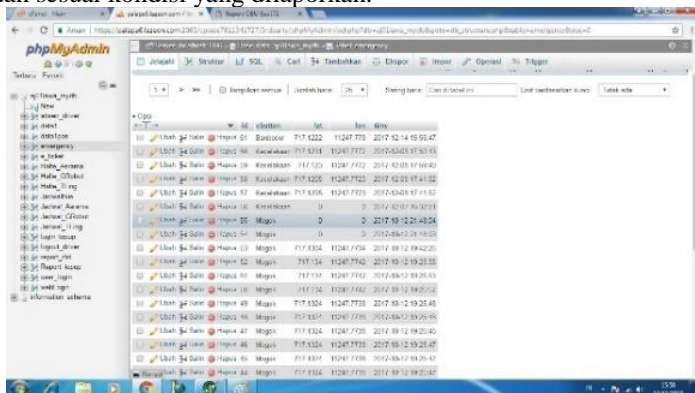
Selain untuk absensi, VMS driver sendiri memiliki fitur lainnya yaitu *emergency button*, apabila armada yang sedang beroperasi mengalami kondisi darurat yang tidak diinginkan seperti ban bocor, penumpang sakit, mogok, dan kecelakaan maka fitur ini akan otomatis memberikan

informasi kepada *server* (*CCROOM*) dan *server* akan segera memberikan bantuan sesuai dengan kondisi darurat apa yang dikirimkan, dan sesuai posisi dari armada yang beroperasi itu berada.



Gambar 4.12 Tampilan Emergency Button

Setelah masuk ke tampilan awal *VMS driver*, dan sudah diklik salah satu kejadian yang dialami oleh bus, maka proses selanjutnya pengiriman informasi ke *server*, dimana proses pengiriman sama dengan pengiriman pesan pada pemantauan armada yang di bahas sebelumnya. Aplikasi akan mengirimkan lokasi dimana armada tersebut berada, kemungkinan *server* menerima informasi dan akan mengirimkan bantuan sesuai kondisi yang dilaporkan.



Gambar 4.13 Report *Emergency Button* di Database

4.1.4 Pengujian *Fitur Ticketing*

Sesuai dengan perancangan yang dijelaskan pada bab sebelumnya, bahwa *fitur ticketing* adalah *fitur* tambahan yang ditempatkan di dalam aplikasi OBU yang sudah dirancang sebelumnya. Di mana dalam *fitur ticketing* ini dapat diuji saat mentab kan RFIDnya, kemudian *top up* tiket, serta tambah saldo dan registrasi member baru. *Fitur e-ticketing* ini bertujuan sebagai alat administrasi bagi para penumpang dalam melakukan perjalanan yang dilakukan dari tujuan satu menuju tujuan lainnya.

1759

COM6 Start Cek Saldo Tambah Saldo Registrasi Member

1 0 binGet

000041D130A0
000041D130A0

Saldo: 20000
NIK: 2213100083
User: mahasiswa
Price: 3000

UPL Upload Top Up
[Http://ad1.bawanghosting.pw/RFID/Upload_Topup.php?cd=000041D130A0&u
UPL Upload User
[Http://ad1.bawanghosting.pw/RFID/Upload_User.php?cd=000041D130A0&u]

Exit

Gambar 4.14 Tampilan Registrasi Member

Sebelum mendapatkan kartu sebagai alat administrasi, para calon penumpang terlebih dahulu wajib melakukan registrasi, dimana didalamnya calon penumpang mengisi data diri seperti nama, NIK, status, dan berapa saldo yang akan di isi lalu tekan *binGet*. Setelah data tersebut di isikan maka identitas penumpang yang mendapatkan kartu tersebut akan otomatis *terecord* kedalam *web report*.

REPORT MEMBER

Print						
No	ID	NIK	USER	HARGA	SALDO	TIME
1	000041D130A0	2213100083	mahasiswa	3000	20000	2017-12-14 15:32:06
2	00004221FC9F	3504030411950003	Mahasiswa	3000	17000	2017-12-14 10:38:46
3	000055A210E7	3214190110940002	Ummu	4000	80000	2017-12-14 10:54:24
4	0000137CA1CE	3013450811970001	Mahasiswa	3000	66000	2017-12-14 10:45:31
5	00001368C4BF	3904010101900002	Ummu	4000	150000	2017-12-13 22:07:50
6	0000136B651D	3109010611920003	Ummu	4000	100000	2017-12-13 22:07:50
7	000017B61CBD	3094170211960003	Mahasiswa	3000	60000	2017-12-14 09:23:45
8	000041743603	0	0	0	0	2017-12-14 15:23:29

Gambar 4.15 Report Registrasi Member Baru

Setelah teregistrasi, pastinya saldo yang ada di dalam tiket akan semakin berkurang, maka dari itu para penumpang wajib menambah saldonya agar dapat melakukan perjalanan kembali.

The screenshot shows a web application interface for managing ticket balances. The interface is divided into several sections:

- Top Bar:** Contains a "1605" label and a "Start" button.
- Navigation Bar:** Includes links for "Cek Saldo", "Tambah Saldo", and "Registrasi Member".
- Form Section:**
 - NIK:** 2213100083
 - Sisa Saldo:** 20000
 - Tambah Saldo:** 10000
 - Jumlah Saldo:** 30000
 - User:** mahasiswa
 - Harga:** 3000
- URLs:**
 - URL Upload User: http://ag01.lavanghosting.pw/RPFD/Upload_User.php?td=000041D130A0&N
 - URL Upload Top Up: http://ag01.lavanghosting.pw/RPFD/Upload_TopUp.php?td=000041D130A0&N
 - URL Identitas: <http://ag01.lavanghosting.pw/RPFD/Identitas.php?td=000041D130A0&N>
 - URL Login: <http://ag01.lavanghosting.pw/RPFD/Login.php?td=000041D130A0&N>
 - URL Logout: <http://ag01.lavanghosting.pw/RPFD/Logout.php?td=000041D130A0&N>
- Buttons:** "Simpan" and "Exit".

Gambar 4.16 Hasil Tambah Saldo

Setelah saldo tiket sudah di tambahkan, maka para calon penumpang dapat menggunakan tiket tersebut untuk melakukan perjalanan ke tempat yang mereka tuju.

4.1.5 Analisa Hasil Pengujian Sistem pada *OBU Side*

Setelah melakukan pengujian, selanjutnya akan dianalisis hasil dari pengujian yang telah dilakukan. Analisa tersebut meliputi apakah sistem yang dibuat tersebut sesuai dengan rancangan yang telah dijabarkan di bab sebelumnya atau tidak. Berikut tabel hasil pengujian analisi sistem pada *OBU side*:

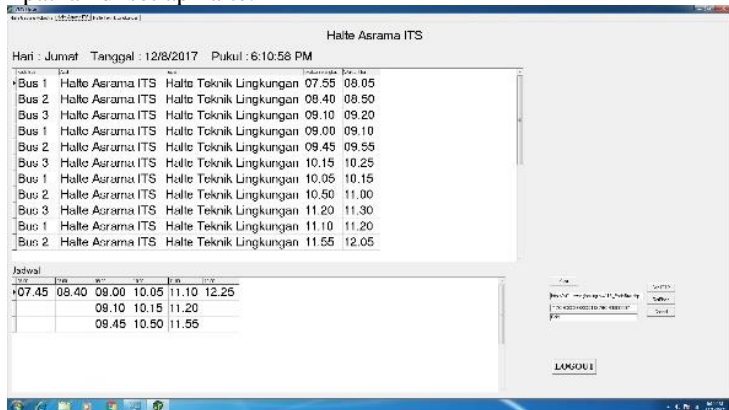
Tabel 4.2 Hasil Pengujian Sistem pada *OBU Side*

NO	Pengujian	Hasil yang di dapat	Analisis
1	Pengiriman Informasi berupa waktu, posisi, dan identitas dari armada menuju <i>server</i>	<i>Server</i> berhasil menerima data dari <i>client</i> (Armada). Pada gambar 4.5	Sistem telah sesuai dengan perancangan
2	Absensi <i>Driver</i>	TAB RFID berfungsi, Identitas sesuai program, dan <i>server</i> berhasil menerima data. Pada gambar 4.11	Sistem telah sesuai dengan perancangan
3	<i>Emergency button</i>	<i>Fitur</i> sudah jadi, <i>server</i> berhasil menerima data. Pada gambar 4.13	Sistem telah sesuai dengan perancangan
4	VMS penumpang	Tujuan selanjutnya sudah dapat ditampilkan pada monitor. Pada gambar 4.8	Sistem telah sesuai dengan perancangan
5	<i>Fitur Ticketing</i>	Top up saldo, Registrasi member, pengiriman data diri member ke <i>web report</i> . Pada gambar 4.14-4.16	Sistem telah sesuai dengan perancangan

6	<i>Button Notification</i>	<i>Box button notification</i> sudah berfungsi dengan baik, dan bisa menimbulkan suara pengingat kepada <i>driver</i>	Sistem telah sesuai dengan perancangan
---	----------------------------	---	--

4.2 Pengujian Sistem pada Halte *Side*

Selain *fitur* yang telah di jelaskan pada penjelasan sebelumnya, sistem yang di buat ini juga termasuk pada halte *side*, yaitu terdapat sebuah mini pc yang didalamnya sudah terinstall aplikasi yang telah dibuat dimana fungsinya adalah untuk memberikan informasi jadwal keberangkatan dan kedatangan kepada calon penumpang, agar calon penumpang dapat meminimalisir waktu menunggu yang lama. Seluruh informasi yang diberikan akan ditampilkan pada monitor yang ditempatkan di setiap halte.



Gambar 4.17 Halte Asrama ITS

Halte Gedung Robotika

Hari : Jumat Tanggal : 12/8/2017 Pukul : 6:09:27 PM

No	Bus	Halte Gedung Robotika	Halte Asrama ITS	Ar. Kedatangan	Ar. Keberangkatan
1	Bus 1	Halte Gedung Robotika	Halte Asrama ITS	07.45	07.55
2	Bus 2	Halte Gedung Robotika	Halte Asrama ITS	08.30	08.40
3	Bus 3	Halte Gedung Robotika	Halte Asrama ITS	09.00	09.10
1	Bus 1	Halte Gedung Robotika	Halte Asrama ITS	08.50	09.00
2	Bus 2	Halte Gedung Robotika	Halte Asrama ITS	09.35	09.45
3	Bus 3	Halte Gedung Robotika	Halte Asrama ITS	10.05	10.15
1	Bus 1	Halte Gedung Robotika	Halte Asrama ITS	09.55	10.05
2	Bus 2	Halte Gedung Robotika	Halte Asrama ITS	10.40	10.50
3	Bus 3	Halte Gedung Robotika	Halte Asrama ITS	11.10	11.20
1	Bus 1	Halte Gedung Robotika	Halte Asrama ITS	11.00	11.10
2	Bus 2	Halte Gedung Robotika	Halte Asrama ITS	11.45	11.55

Jadwal

07.00	08.00	09.00	10.00	11.00	12.00
07.45	08.30	09.00	10.05	11.00	12.25
	08.50	09.35	10.40	11.10	
		09.55		11.45	

LOGOUT

Gambar 4.18 Halte Robotika

Halte Teknik Lingkungan

Hari : Jumat Tanggal : 12/8/2017 Pukul : 6:11:33 PM

No	Bus	Halte Teknik Lingkungan	Halte Gedung Robotika	Ar. Kedatangan	Ar. Keberangkatan
1	Bus 1	Halte Teknik Lingkungan	Halte Gedung Robotika	08.05	08.35
2	Bus 2	Halte Teknik Lingkungan	Halte Gedung Robotika	08.50	09.20
3	Bus 3	Halte Teknik Lingkungan	Halte Gedung Robotika	09.20	09.50
1	Bus 1	Halte Teknik Lingkungan	Halte Gedung Robotika	09.10	09.40
2	Bus 2	Halte Teknik Lingkungan	Halte Gedung Robotika	09.55	10.25
3	Bus 3	Halte Teknik Lingkungan	Halte Gedung Robotika	10.25	10.55
1	Bus 1	Halte Teknik Lingkungan	Halte Gedung Robotika	10.15	10.45
2	Bus 2	Halte Teknik Lingkungan	Halte Gedung Robotika	11.00	11.30
3	Bus 3	Halte Teknik Lingkungan	Halte Gedung Robotika	11.30	12.00
1	Bus 1	Halte Teknik Lingkungan	Halte Gedung Robotika	11.20	11.50
2	Bus 2	Halte Teknik Lingkungan	Halte Gedung Robotika	12.05	12.35
3	Bus 3	Halte Teknik Lingkungan	Halte Gedung Robotika	12.35	13.05

Jadwal

08.05	09.10	10.15	11.00	12.05
08.05	09.20	10.25	11.20	12.35
	09.55		11.30	

LOGOUT

Gambar 4.19 Halte Teknik Lingkungan

Pada awalnya direncanakan menggunakan tiga halte di daerah seputaran kampus ITS ketiga halte tersebut pada halte asrama ITS, gedung robotika, dan Teknik Lingkungan.

4.2.1 Analisa Hasil Pengujian Sistem pada Halte Side

Setelah melakukan pengujian, selanjutnya akan dianalisis hasil dari pengujian yang telah dilakukan. Analisa tersebut meliputi apakah sistem yang di buat tersebut sesuai dengan rancangan yang telah dijabarkan di bab sebelumnya atau tidak.

Tabel 4.3 Hasil Pengujian Sistem pada Halte *Side*

NO	Pengujian	Hasil yang di dapat	Analisis
1	Jadwal Keberangkatan dan kedatangan armada pada halte	Jadwal dapat ditampilkan	Sistem telah sesuai dengan perancangan

4.3 Pengujian dan Analisa Jaringan Komunikasi Sistem

Setelah *prototype* telah selesai, dan beberapa *fitur* sudah bisa di operasikan, untuk selanjutnya pengujian jaringan komunikasi sistem ini bertujuan untuk mendapatkan ukuran data terbaik dalam pengiriman paket data dari *client* menuju ke *server* dalam arti pengiriman pesan dari armada yang sedang beroperasi menuju *database* atau *webreport*.

Pengujian jaringan ini dilakukan dengan cara mengirimkan data selama 5 menit secara terus menerus tanpa terputus. Untuk mengetahui nilai serta parameter QOS dapat menggunakan *software Wireshark* untuk dapat menemukan parameter pendukung agar dapat mengetahui seberapa besar nilai *throughput* dan *delay* yang di dapat dalam proses pengiriman informasi dari OBU yang berada di tiap bus menuju ke *server*.

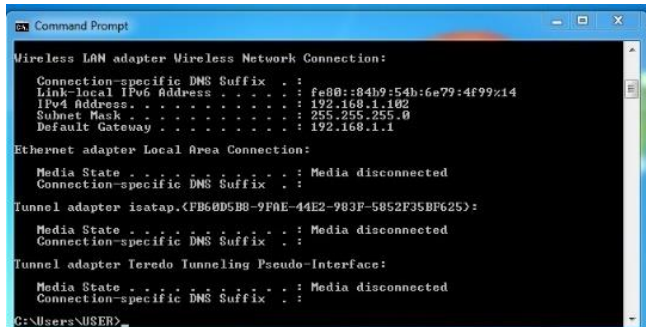
Pengukuran di lakukan dalam 8 hari dari hari Kamis 04 Januari sampai Kamis 11 Januari dengan jangka waktu 5 menit, pada pukul 17.00 WIB mengambil 30 sampel paket data dengan parameter Panjang *bits*, dan waktu yang sama untuk di amati dan di analisa apakah jaringan yang di pakai memiliki *throughput* yang baik dan *delay* yang rendah atau tidak.

4.3.1 Pengujian dan Analisa Throughput

Pada pengujian *throughput* ini digunakan untuk mendapatkan jumlah Panjang paket data yang optimal dalam proses pengiriman data dari OBU menuju *server*. Pengujian ini dilakukan dengan cara mengirimkan paket data secara terus menerus selama 5 menit tanpa terputus, mengambil waktu selama 8 hari diambil sore hari pada pukul 17.00, kemudian mengambil 30 sampel dengan parameter Panjang bit yang sama dan jam yang sama, kemudian mengambil nilai rata-rata dari 30 sampel yang didapat, untuk mendapatkan nilai rata-rata *throughput* pada setiap harinya. Kemudian di bandingkan dengan hari-hari lainnya.

Pertama yang harus dilakukan adalah dengan mengecek ip address dari Komputer yang kita gunakan, komputer yang digunakan ini dalam

arti adalah OBU dan sudah terinstall *software wireshark*. Setelah mengecek IP komputer sendiri, kemudian mencoba mengecek IP dari server yang digunakan yaitu *aj01.lawanghosting.pw*.



```
Command Prompt

Wireless LAN adapter Wireless Network Connection:
    Connection-specific DNS Suffix . : 
    Link-local IPv6 Address . . . . . : fe80::84b9:54b:6e79:4f99%14
    IPv4 Address. . . . . : 192.168.1.102
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.1.1

Ethernet adapter Local Area Connection:
    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix . : 

Tunnel adapter isatap.{FB60D5B8-9FAE-44E2-903F-5852F35BF625}:
    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix . : 

Tunnel adapter Teredo Tunneling Pseudo-Interface:
    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix . : 

C:\Users\USER>
```

Gambar 4.20 Cek IP dengan *Command Prompt*

Setelah dicek ip nya, lalu mencoba untuk *ping database* atau *server* yang digunakan dalam media penyimpanan informasi, *ping* ke alamat *aj01.lawanghosting.pw*, dan mendapatkan ip dari *server* tersebut untuk lebih memudahkan mendapatkan parameter untuk perhitungan dari kualitas jaringan yang akan di analisa.



```
Command Prompt

Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\USER>ping aj01.lawanghosting.pw

Pinging aj01.lawanghosting.pw [103.27.206.17] with 32 bytes of data:
Reply from 103.27.206.17: bytes=32 time=70ms TTL=53
Reply from 103.27.206.17: bytes=32 time=78ms TTL=53
Reply from 103.27.206.17: bytes=32 time=68ms TTL=53
Reply from 103.27.206.17: bytes=32 time=67ms TTL=53

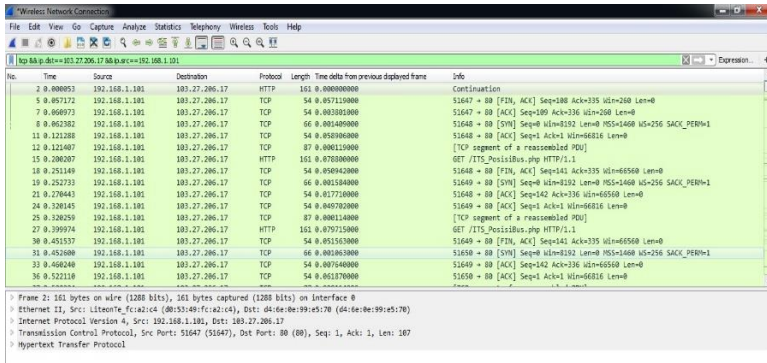
Ping statistics for 103.27.206.17:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 67ms, Maximum = 78ms, Average = 70ms

C:\Users\USER>
```

Gambar 4.21 Ping Server Menggunakan *Command Prompt*

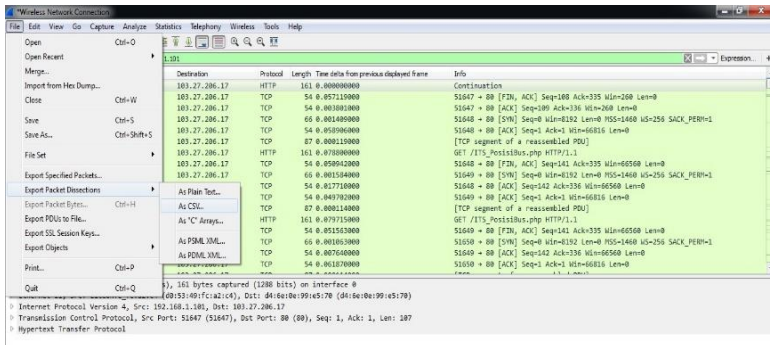
Setelah server berhasil di ping, untuk selanjutnya masuk ke *software wireshark*, dan menganalisa parameter dari kualitas jaringan yang akan diamati yaitu mengecek Panjang *bits* serta waktu tempuh dalam

pengiriman informasi setiap pakatnya, dengan cara menjalankan *wireshark* lalu *capture* hasil pengiriman informasi yang dikirimkan dari OBU menuju *server*.



Gambar 4.22 Tampilan *Capture* di *Wireshark*

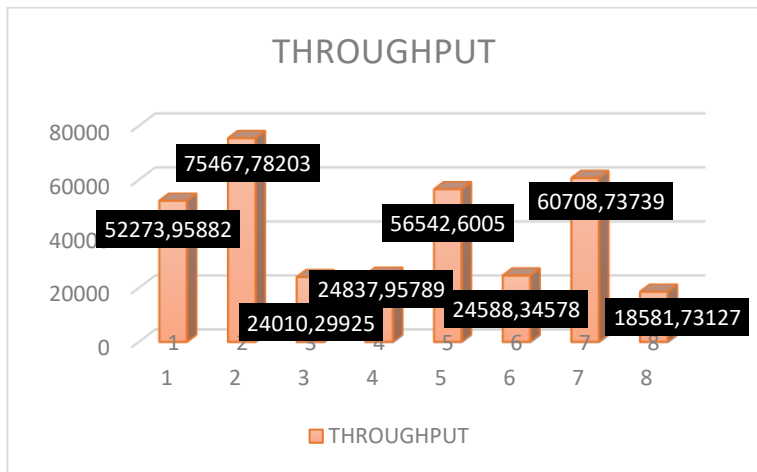
Setelah mengcapture hasil pengiriman informasi selama 5 menit, selanjutnya *filter* pengiriman informasi dari OBU menuju *server* untuk menampilkan hasil yang di amati saja, dengan cara ketik "tcp && ip.dst==192.168.1.101 && ip.src==103.27.206.17" lalu tekan *enter* dan kemudian *wireshark* akan menampilkan hasil pengiriman informasi yang dikirimkan dari OBU menuju *server*.



Gambar 4.23 Tampilan *Save CSV* pada *Wireshark*

Setelah mem *filter* informasi yang menjadi parameter pengukuran, kemudian simpan hasil pengukuran tersebut dalam format CSV kemudian dimasukkan ke *excel*, setelah itu dianalisa dan didapatkan hasil analisa *throughput* rata-rata nya.

Setelah melakukan pengukuran selama 7 hari, didapatkan lah hasil pengukuran yang telah dilakukan sebelumnya, dan hasil bisa dilihat pada grafik yang ditampilkan di bawah ini:



Gambar 4.24 Grafik Pengujian *Throughput*

Dari pengujian *throughput* selama 8 hari berturut-turut dimulai pada hari Kamis 4 Januari hingga hari Kamis 11 Januari, didapatkan hasil *throughput* di mana dapat di analisa bahwa nilai *throughput* yang dimiliki sistem ini pada kondisi yang naik turun. Pola tersebut digambarkan dari nilai *throughput* yang dihasilkan. Penyebab terjadinya naik turun tersebut karena kondisi sinyal yang tidak stabil. Namun dari ke delapan hari pengukuran tersebut didapatkan nilai *throughput* terbesar berada di hari ke 2 yaitu dengan nilai 75457,78 bps dan nilai terkecil berada di hari ke 8 yaitu dengan nilai 18581,731 bps.

4.3.2 Pengujian dan Analisa Delay

Sama halnya dengan pengujian *throughput*, pengujian *delay* pun menggunakan *software wireshark* dengan metode pengujian yang sama yaitu mengirimkan paket data dari OBU menuju *server* dalam jangka waktu 5 menit tanpa berhenti melakukan pengujian selama 8 hari pada jam 17.00 di setiap harinya dimulai pada hari kamis tanggal 4 Januari hingga Kamis 11 Januari 2018 bertempat di lab AJ404.

Dari proses pengiriman tersebut diambil 30 sampel data yang diolah dengan mengambil parameter waktu saat paket diterima dan waktu paket dikirimkan untuk menemukan nilai *delay*. Proses pengambilan data untuk *delay* ini pun sama dengan pengujian pada *throughput* dengan mengecek *ip address* dari Komputer yang kita gunakan, Komputer yang digunakan ini dalam arti adalah OBU yang sudah ter *install software wireshark*.



```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\USER>ping aj01.lawanghosting.pw

Pinging aj01.lawanghosting.pw [103.27.206.17] with 32 bytes of data:
Reply from 103.27.206.17: bytes=32 time=70ms TTL=53
Reply from 103.27.206.17: bytes=32 time=78ms TTL=53
Reply from 103.27.206.17: bytes=32 time=68ms TTL=53
Reply from 103.27.206.17: bytes=32 time=67ms TTL=53

Ping statistics for 103.27.206.17:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 67ms, Maximum = 78ms, Average = 70ms

C:\Users\USER>
```

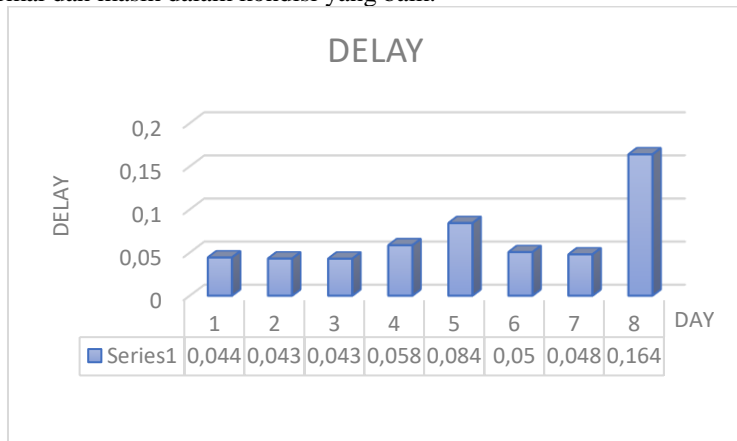
Gambar 4.25 Ping Server Menggunakan Command Prompt

Setelah mengetahui *ip address* dari komputer yang sudah di *install wireshark* dan mengetahui *ip address* dari *server aj01.lawanghosting.pw* langkah selanjutnya yaitu *ping server* menggunakan *Command Prompt*, setelah berhasil ter *ping* masuk ke *wireshark* yang sudah terinstall, kemudian *capture* pada jaringan seluler dan didapatkan lah data yang di perlukan. Setelah mendapatkan data yang di perlukan *save* dengan dormat *csv*, lalu buka hasil *csv* tersebut di *excel* dan dianalisa parameter yang didapatkan untuk menentukan nilai *delay*.

No.	Time	Source	Destination	Protocol	Length	Time delta from previous displayed frame	Info
2	0.000053	192.168.1.101	183.27.206.17	HTTP	161	0.000000000	Continuation
3	0.001732	192.168.1.101	183.27.206.17	TCP	54	0.001679000	51647 → 80 [FIN, ACK] Seq=388 Ack=335 Win=240 Len=0
7	0.006973	192.168.1.101	183.27.206.17	TCP	54	0.003301000	51647 → 80 [ACK] Seq=180 Ack=336 Win=256 Len=0
8	0.062382	192.168.1.101	183.27.206.17	TCP	66	0.001409000	51648 → 80 [SYN] Seq=0 Win=0 Len=0 MSS=1460 WS=256 SACK_PERM=1
11	0.121288	192.168.1.101	183.27.206.17	TCP	54	0.058906000	51648 → 80 [ACK] Seq=1 Ack=1 Win=6616 Len=0
12	0.122487	192.168.1.101	183.27.206.17	TCP	87	0.000120000	[TCP segment of a reassembled PDU]
15	0.206207	192.168.1.101	183.27.206.17	HTTP	161	0.073800000	GET /ITS_Pesilibus.php HTTP/1.1
18	0.251149	192.168.1.101	183.27.206.17	TCP	54	0.050942000	51648 → 80 [FIN, ACK] Seq=141 Ack=335 Win=66560 Len=0
19	0.252733	192.168.1.101	183.27.206.17	TCP	66	0.001584000	51649 → 80 [SYN] Seq=0 Win=0 Len=0 MSS=1460 WS=256 SACK_PERM=1
21	0.278643	192.168.1.101	183.27.206.17	TCP	54	0.027709000	51648 → 80 [ACK] Seq=142 Ack=336 Win=66560 Len=0
24	0.328145	192.168.1.101	183.27.206.17	TCP	54	0.049703000	51649 → 80 [ACK] Seq=1 Ack=1 Win=6616 Len=0
25	0.328259	192.168.1.101	183.27.206.17	TCP	87	0.000114000	[TCP segment of a reassembled PDU]
27	0.399374	192.168.1.101	183.27.206.17	HTTP	161	0.070750000	GET /ITS_Pesilibus.php HTTP/1.1
30	0.451537	192.168.1.101	183.27.206.17	TCP	54	0.051533000	51649 → 80 [FIN, ACK] Seq=141 Ack=335 Win=66560 Len=0
31	0.452680	192.168.1.101	183.27.206.17	TCP	66	0.001463000	51650 → 80 [SYN] Seq=0 Win=0 Len=0 MSS=1460 WS=256 SACK_PERM=1
33	0.468240	192.168.1.101	183.27.206.17	TCP	54	0.007540000	51649 → 80 [ACK] Seq=142 Ack=336 Win=66560 Len=0
36	0.522118	192.168.1.101	183.27.206.17	TCP	54	0.051170000	51650 → 80 [ACK] Seq=1 Ack=1 Win=6616 Len=0

Gambar 4.26 Tampilan Data di Wireshark

Dari pengujian *delay* selama 8 hari berturut-turut dimulai di hari kamis 4 Januari hingga hari kamis 11 Januari, didapatkan hasil *delay* di mana dapat dianalisa bahwa nilai *delay* yang dimiliki sistem ini memberikan nilai yang naik turun, pola tersebut menggambarkan dari nilai *delay* yang dihasilkan pada gambar 4.27. Penyebab terjadinya pola tersebut bisa dikarenakan karena kondisi jaringan yang tidak stabil, atau trafik pengiriman yang besar sehingga menimbulkan *delay* yang naik turun, namun nilai *delay* yang dimiliki tersebut masih tergolong batas normal dan masih dalam kondisi yang baik.



Gambar 4.27 Grafik Pengujian Delay

Dari hasil pengujian delay yang telah dilakukan tersebut, dapat dianalisis bahwa nilai *delay* terbesar yaitu berada pada hari ke 8 Kamis 11 Januari dimana berada dinilai 0,1638385 s atau 163,8385 ms. Sedangkan nilai *delay* terkecil berada di hari ketiga yaitu pengukuran pada tanggal 6 Januari 2018 dimana berada di nilai 0,0429494 s atau senilai dengan 42,9494 ms. Dari ke delapan pengukuran yang telah dilakukan sebelumnya bisa dilihat bahwa nilai delay berada pada kondisi yang normal sesuai dengan standar ITU-T *recommendation* Y1541 Dimana nilai *delay* <150 ms berada pada kondisi yang *excellent*.

[Halaman ini sengaja dikosongkan]

BAB V

PENUTUP

5.1 Kesimpulan

Setelah dilakukan perancangan dan pengujian sistem dan jaringan komunikasi terintegrasi Angkutan Massal Cepat Surabaya, maka secara keseluruhan dapat diambil kesimpulan sebagai berikut:

1. *Prototype On Board Unit* dan *Variable Messaging System* sudah dapat direalisasikan kemudian dapat dioperasikan juga setiap *fitur-fitur* yang ada didalamnya.
2. Pengiriman data dari *On Board Unit* menuju *server* dan *web report* pun, sudah dapat dilakukan dengan sesuai tanpa ada nya data yang tidak akurat.
3. Nilai *throughput* terbesar berada di hari ke 2 Jumat tanggal 5 Januari dengan nilai 75457,78 bps dan nilai terkecil berada pada hari ke 8 Kamis 11 Januari yaitu dengan nilai 18581,731 bps.
4. Nilai *delay* terbesar yaitu berada pada hari ke 8 Kamis 11 Januari dimana berada dinilai 0,1638385 s atau 163,8385 ms. Sedangkan nilai *delay* terkecil berada di hari ketiga yaitu pengukuran pada tanggal 6 Januari 2018 dimana berada di nilai 0,0429494 s atau senilai dengan 42,9494 ms, di mana dalam standarisasi ITU-T nilai *delay* di bawah dari 150 ms (<150ms) tergolong dalam kondisi yang *excellent*.

5.2 Saran

Adapun hal-hal yang masih bisa dikembangkan dari tugas akhir yang dibuat ini adalah:

1. Pembuatan untuk penjadwalan yang dapat di *update* secara otomatis di setiap halte
2. Pengembangan *prototype* dengan memasukkan *fitur-fitur* baru didalamnya, agar lebih mendukung kecanggihan dari *prototype* tersebut
3. Pengukuran serta penggunaan *provider* jaringan yang bervariasi, agar mendapatkan hasil yang maksimal dalam melakukan pengiriman serta penerimaan informasi dari *client-server*

[Halaman ini sengaja dikosongkan]

DAFTAR PUSTAKA

- [1] Affandi, Achmad dan tim. “Laporan Akhir Kajian Sistem dan Teknologi IT dalam Rangka Integrasi Angkutan Massal Cepat *Trunk* dan *Feeder* AMC Surabaya,” Dinas Perhubungan Kota Surabaya, Surabaya, 2015.
- [2] Advanced Public Transportation Systems: The State of the Art Update 2000, Federal Transit Administration Office of Research, Demonstration and Innovation Office of Mobility Innovation Advanced Public Transportation Systems Division.
- [3] Jedok Kim, Young-Jun Moon and In-soo Suh, IEEE SMART MOBILITY STRATEGY IN KOREA on Sustainability, Safety and Efficiency Toward 2025.
- [4] Haroon Shakirat Oluwatosin, IOSR Journal of Komputer Engineering (IOSR-JCE) *School of Computing Universiti Utara Malaysia Kedah, Malaysia*.
- [5] Eko Setijadi S.T, M.T. Ph. D; SMART-CITY & Intelligent Solution TEAM LPPM ITS 2015, Laporan Akhir Detail Engineering Design Manajemen Armada Angkutan Massal Cepat Surabaya.
- [6] DARIO HIDALGO Senior Transport Engineer EMBARQ, The WRI Center for Sustainable Transport, *Lessons learned from major bus improvements in Latin America and Asia, Modernizing Public Transportation*.
- [7] Sastriyana, Nym Yuni. “Perancangan dan Analisis Interkoneksi Jaringan pada *Electronic Toll Collection* Berbasis *Radio Frequency Identification* (RFID),” Institut Teknologi Sepuluh Nopember, Surabaya, 2010.
- [8] ITU-T Series G, “*Transmission system and media, digital system and network*,” 2003.
- [9] Faesal, Andris. “Pemrograman *Client-server* dengan Borland Delphi 7 dan My SQL,” STMIK Bumigora, Mataram, 2012
- [10] Yuliantoro, Prasetyo. “Rancang Bangun Protokol *E-Ticketing*,” Institut Teknologi Sepuluh Nopember, Surabaya, 2015

[Halaman ini sengaja dikosongkan]

LAMPIRAN

A. Program Delphi 6

1. ABSENSI DRIVER

```
unit Unit1;
```

```
interface
```

```
uses
```

```
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,  
  Forms,
```

```
  Dialogs, StdCtrls, CPort, IdBaseComponent, IdComponent,  
  IdTCPConnection,
```

```
  IdTCPClient, IdHTTP, ExtCtrls;
```

```
type
```

```
  TForm1 = class(TForm)
```

```
    Label1: TLabel;
```

```
    Label2: TLabel;
```

```
    Label3: TLabel;
```

```
    Edit2: TEdit;
```

```
    Edit3: TEdit;
```

```
    Memo2: TMemo;
```

```
    Button3: TButton;
```

```
    Button4: TButton;
```

```
    Memo1: TMemo;
```

```
    EditComPort: TEdit;
```

```
    btnStart: TButton;
```

```
    pnlWaitingData: TPanel;
```

```
    pnlWaitingDataEx: TPanel;
```

```
    EditRfid: TEdit;
```

```
    Timer1: TTimer;
```

```
    IdHTTP1: TIdHTTP;
```

```
    Button1: TButton;
```

```
    procedure Button3Click(Sender: TObject);
```

```
    procedure Button4Click(Sender: TObject);
```

```
    procedure btnStartClick(Sender: TObject);
```

```
    procedure Timer1Timer(Sender: TObject);
```

```

    procedure Button1Click(Sender: TObject);

private
    { Private declarations }
public
    { Public declarations }
end;

var
    Form1: TForm1;

implementation

Uses SynaSer, Unit2;

{$R *.dfm}

var
    ser:TBlockSerial;

procedure TForm1.btnStartClick(Sender: TObject);
begin
    (sender as TButton).Enabled:=False;
    ser:=TBlockSerial.Create;
    ser.RaiseExcept:=False;
    try
        ser.Connect(EditComPort.Text);
        ser.Config(9600,8,'N',0,false,false);
    finally
        end;
    Timer1.Enabled:=True;
end;

var
    xCCounter:Word;

procedure TForm1.Timer1Timer(Sender: TObject);

```

```

var
  s:ShortString;
begin
  inc(xCounter);
  pnlWaitingData.Caption:=IntToStr(Ser.WaitingData);
  pnlWaitingDataEx.Caption:=IntToStr(Ser.WaitingDataEx);
  Caption:=IntToStr(xCounter);
  if Ser.WaitingData>=16 then
    begin
      s:=ser.Recvstring(10);
      if s[1]<#32 then delete(s,1,1);
      if s[1]<#32 then delete(s,1,1);
      EditRfid.Text:=s;
      Memo1.Lines.Add(s);
    begin
      if pos('7800278E4293',s)>0 then
        begin
          Edit2.Text:='AdiSupono';
        end
      else if pos('6F0085BFFFAA',s)>0 then
        begin
          Edit2.Text:='SlametHariyadi';
        end
      else if pos('780027BBAC48',s)>0 then
        begin
          Edit2.Text:='SamuelWicaksono';
        end
      else if pos('6F0085BFECB9',s)>0 then
        begin
          Edit2.Text:='MustofaAli';
        end
      end;
    Button3.Click;
    end;
  end;
end;

```

```

procedure TForm1.Button3Click(Sender: TObject);
var

```

```

refURL:ShortString;
s:String;
begin
refURL:='http://aj01.lawanghosting.pw/absen_driver.php';
Edit3.Text:=
refURL+
'?nama='+Edit2.Text+'&xtag='+EditRfid.Text;
s:=IdHttp1.Get(Edit3.Text);
Memo2.Text:=s;
end;

procedure TForm1.Button4Click(Sender: TObject);
begin
Application.Terminate;
end;

procedure TForm1.Button1Click(Sender: TObject);
begin
Form2.Visible:=true;
Form1.Visible:=false;
end;

end.

```

2. VMS PENUMPANG

```

unit Unit2;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms,
  Dialogs, StdCtrls, DB, ZAbstractRODataset, ZAbstractDataset,
  ZDataset,
  ZConnection;

type

```



```

TForm2 = class(TForm)
  Label1: TLabel;
  Edit1: TEdit;
  Edit2: TEdit;
  Button1: TButton;
  Button2: TButton;
  Label2: TLabel;
  Label3: TLabel;
  ZConnection1: TZConnection;
  ZQuery1: TZQuery;
  CheckBox1: TCheckBox;
  procedure Button1Click(Sender: TObject);
  procedure Button2Click(Sender: TObject);
  procedure CheckBox1Click(Sender: TObject);
  procedure FormCreate(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form2: TForm2;

implementation

uses Unit1, Unit3, Unit7, Unit4, Unit6, Unit8, Unit9;

{$R *.dfm}

procedure TForm2.Button1Click(Sender: TObject);
begin
  with zquery1 do begin
    SQL.Clear;
    SQL.Add('select * from user_log where
    username='+QuotedStr(edit1.Text));
    open;
    end;
  //end with

```

```

//jika tidak ditemukan data yang dicari
//maka tampilkan pesan
if ZQuery1.RecordCount=0
then
Application.MessageBox('Maaf user name tidak
ditemukan','informasi',MB_OK or MB_ICONINFORMATION)
else
begin
if ZQuery1.FieldByName('password').AsString<>Edit2.Text
then
Application.MessageBox('Pastikan password yang anda masukkan
benar','error',MB_OK or MB_ICONERROR)
else
begin
hide;
Form7.Show;
Form4.Left:=-1368;
Edit1.clear;
Edit2.clear;
end;
end;
end;

procedure TForm2.Button2Click(Sender: TObject);
begin
application.Terminate
end;

procedure TForm2.CheckBox1Click(Sender: TObject);
begin
if CheckBox1.Checked then
begin
edit2.PasswordChar:=#0;
end
else
begin
edit2.PasswordChar:='*';
end;
end;
end;

```

```

procedure TForm2.FormCreate(Sender: TObject);
begin
  Edit2.PasswordChar:='*';
end;

end.

```

3. TICKETING SYSTEM

```

unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms,
  Dialogs, StdCtrls, ExtCtrls, Gauges, ComCtrls, IdBaseComponent,
  IdComponent, IdTCPConnection, IdTCPClient, IdHTTP;

type
  TForm1 = class(TForm)
    Memo1: TMemo;
    Timer1: TTimer;
    EditComPort: TEdit;
    btnStart: TButton;
    pnlWaitingData: TPanel;
    pnlWaitingDataEx: TPanel;
    EditUrl: TEdit;
    EditRfid: TEdit;
    btnGet: TButton;
    IdHTTP1: TIdHTTP;
    Edit1: TEdit;
    Edit2: TEdit;
    Edit3: TEdit;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;

```

```

Edit4: TEdit;
IdHTTP2: TIdHTTP;
Memo2: TMemo;
Memo3: TMemo;
Label5: TLabel;
Label6: TLabel;
Button1: TButton;
Button2: TButton;
Button3: TButton;
Edit5: TEdit;
Label7: TLabel;
Edit6: TEdit;
Memo4: TMemo;
Label8: TLabel;
IdHTTP3: TIdHTTP;
procedure FormCreate(Sender: TObject);
procedure btnStartClick(Sender: TObject);
procedure Timer1Timer(Sender: TObject);
procedure btnGetClick(Sender: TObject);
procedure Button1Click(Sender: TObject);
procedure Button2Click(Sender: TObject);
procedure Button3Click(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }
end;

var
    Form1: TForm1;

implementation

Uses SynaSer, Unit2, Unit3;

{$R *.dfm}

var

```

```

ser:TBlockSerial;

procedure TForm1.FormCreate(Sender: TObject);
begin
  Caption:='AT-Cmd2016';
end;

procedure TForm1.btnStartClick(Sender: TObject);
begin
  (sender as TButton).Enabled:=False;
  ser:=TBlockSerial.Create;
  ser.RaiseExcept:=False;
  try
    ser.Connect(EditComPort.Text);
    ser.Config(9600,8,'N',0,false,false);
  finally
    end;
  Timer1.Enabled:=True;
end;

var
  xCounter:Word;

procedure TForm1.Timer1Timer(Sender: TObject);
var
  s:ShortString;
begin
  inc(xCounter);
  pnlWaitingData.Caption:=IntToStr(Ser.WaitingData);
  pnlWaitingDataEx.Caption:=IntToStr(Ser.WaitingDataEx);
  Caption:=IntToStr(xCounter);
  if Ser.WaitingData>=16 then
  begin
    s:=ser.Recvstring(10);
    if s[1]<#32 then delete(s,1,1);
    if s[1]<#32 then delete(s,1,1);
    EditRfid.Text:=s;
  end;
end;

```

```
EditUrl.Text:='Http://aj01.lawanghosting.pw/RFID/Uploud_Saldo.php?id='+EditRfid.Text+'&balance='+Edit1.Text;
```

```
Edit4.Text:='Http://aj01.lawanghosting.pw/RFID/Uploud_User.php?id='+EditRfid.Text+'&NIK='+Edit5.Text+'&user='+Edit2.Text+'&price='+Edit3.Text;
```

```
Edit6.Text:='Http://aj01.lawanghosting.pw/RFID/Uploud_Topup.php?id='+EditRfid.Text+'&nik='+Edit5.Text+'&user='+Edit2.Text+'&harga='+Edit3.Text+'&saldo='+Edit1.Text;
```

```
    Memo1.Lines.Add(s);  
    btnGet.Click;  
end;  
end;
```

```
procedure TForm1.btnGetClick(Sender: TObject);  
var  
    sResp,user,topup:String;  
begin  
    sResp:=idHttp1.Get(EditUrl.Text);  
    Memo2.Lines.Add(sResp);  
    user:=idHttp2.Get(Edit4.Text);  
    Memo3.Lines.Add(user);  
    topup:=idHttp3.Get(Edit6.Text);  
    Memo4.Lines.Add(topup);  
end;
```

```
procedure TForm1.Button1Click(Sender: TObject);  
begin  
    Form2.Visible:=True;  
    Form1.Visible:=False;  
    Form3.Visible:=False;  
end;
```

```
procedure TForm1.Button2Click(Sender: TObject);  
begin  
    application.Terminate;  
end;
```

```

procedure TForm1.Button3Click(Sender: TObject);
begin
Form2.Visible:=False;
Form1.Visible:=False;
Form3.Visible:=True;
end;

end.

```

4. HALTE

```

unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms,
  Dialogs, DB, ZAbstractRODataset, ZAbstractDataset, ZDataset,
  ZConnection,
  StdCtrls;

type
  TForm1 = class(TForm)
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Edit1: TEdit;
    Edit2: TEdit;
    Button1: TButton;
    Button2: TButton;
    CheckBox1: TCheckBox;
    ZConnection1: TZConnection;
    ZQuery1: TZQuery;
    procedure Button1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    procedure CheckBox1Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
  private

```

```

    { Private declarations }
public
    { Public declarations }
end;

var
    Form1: TForm1;

implementation

uses Unit2, Unit5;

{$R *.dfm}

procedure TForm1.Button1Click(Sender: TObject);
begin
    with zquery1 do begin
        SQL.Clear;
        SQL.Add('select * from user_login where
        username='+QuotedStr(edit1.Text));
        open;
        end;
    //end with
    //jika tidak ditemukan data yang dicari
    //maka tampilkan pesan
    if ZQuery1.RecordCount=0
    then
        Application.MessageBox('Maaf user name tidak
        ditemukan','informasi',MB_OK or MB_ICONINFORMATION)
    else
        begin
            if ZQuery1.FieldName('password').AsString<>Edit2.Text
            then
                Application.MessageBox('Pastikan password yang anda masukkan
                benar','error',MB_OK or MB_ICONERROR)
            else
                begin
                    hide;
                    //form2.show;

```



```
Form5.Show;  
Edit1.Clear;  
Edit2.Clear;  
end;  
end;  
end;
```

```
procedure TForm1.Button2Click(Sender: TObject);  
begin  
  application.Terminate  
end;
```

```
procedure TForm1.CheckBox1Click(Sender: TObject);  
begin  
  if CheckBox1.Checked then  
  begin  
    edit2.PasswordChar:=#0;  
  end  
  else  
  begin  
    edit2.PasswordChar:='*';  
  end;  
end;
```

```
procedure TForm1.FormCreate(Sender: TObject);  
begin  
  Edit2.PasswordChar:='*';  
end;
```

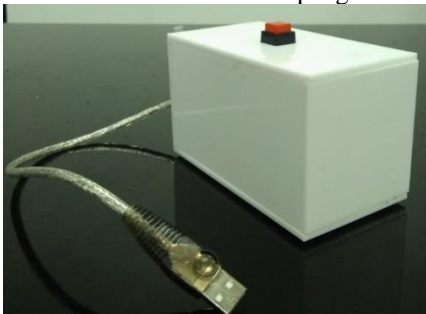
```
end.
```

B. PERANGKAT *On Board Unit* (OBU) LENGKAP

1. Prototype *On Board Unit* (OBU)



2. *Box Button* untuk Penumpang Turun



3. Tag RFID Ticketing



4. Integrated Power Supply sistem OBU



C. TABEL HASIL PENGUJIAN JARINGAN

1. Pengujian *Throughput* dan *Delay* Hari 1 Kamis 4 Januari

No	<i>Length</i>	<i>bits</i>	<i>Delay</i>	ms	<i>Throughput</i>
1	54	432	0,075929	75,929	5689,525741
2	54	432	0,008636	8,636	50023,15887
3	54	432	0,056063	56,063	7705,616895
4	54	432	0,073544	73,544	5874,034592
5	54	432	0,00046	0,46	939130,4348
6	54	432	0,075214	75,214	5743,611562
7	54	432	0,053885	53,885	8017,073397
8	54	432	0,006519	6,519	66267,83249
9	54	432	0,057959	57,959	7453,544747
10	54	432	0,059883	59,883	7214,067431
11	54	432	0,027108	27,108	15936,25498
12	54	432	0,068476	68,476	6308,779718
13	54	432	0,053549	53,549	8067,377542
14	54	432	0,009311	9,311	46396,73504
15	54	432	0,056063	56,063	7705,616895
16	54	432	0,049516	49,516	8724,452702
17	54	432	0,009325	9,325	46327,07775
18	54	432	0,056255	56,255	7679,317394
19	54	432	0,065049	65,049	6641,147443
20	54	432	0,005292	5,292	81632,65306
21	54	432	0,072472	72,472	5960,922839
22	54	432	0,053043	53,043	8144,335728
23	54	432	0,018518	18,518	23328,6532
24	54	432	0,047263	47,263	9140,34234

25	54	432	0,069972	69,972	6173,898131
26	54	432	0,003793	3,793	113894,0153
27	54	432	0,057158	57,158	7557,997131
28	54	432	0,071992	71,992	6000,666741
29	54	432	0,01034	10,34	41779,4971
30	54	432	0,056103	56,103	7700,122988

2. Pengujian *Throughput* dan *Delay* Hari 2 Jumat 5 Januari

No.	Length	bits	Delay	ms	Throughput
1	54	432	0,058275	58,275	7413,127413
2	54	432	0,008723	8,723	49524,24625
3	54	432	0,060126	60,126	7184,911685
4	54	432	0,070135	70,135	6159,54944
5	54	432	0,001162	1,162	371772,8055
6	54	432	0,068862	68,862	6273,416398
7	54	432	0,065811	65,811	6564,252177
8	54	432	0,005571	5,571	77544,42649
9	54	432	0,061504	61,504	7023,933403
10	54	432	0,061393	61,393	7036,632841
11	54	432	0,009202	9,202	46946,31602
12	54	432	0,048111	48,111	8979,235518
13	54	432	0,060953	60,953	7087,428018
14	54	432	0,006874	6,874	62845,5048
15	54	432	0,058536	58,536	7380,073801
16	54	432	0,051083	51,083	8456,825167
17	54	432	0,010805	10,805	39981,49005
18	54	432	0,071999	71,999	6000,083334
19	54	432	0,05507	55,07	7844,561467
20	54	432	0,003427	3,427	126057,7765

21	54	432	0,060021	60,021	7197,480882
22	54	432	0,062772	62,772	6882,049321
23	54	432	0,000349	0,349	1237822,35
24	54	432	0,056348	56,348	7666,643004
25	54	432	0,07151	71,51	6041,113131
26	54	432	0,005855	5,855	73783,09137
27	54	432	0,059133	59,133	7305,56542
28	54	432	0,059954	59,954	7205,524235
29	54	432	0,012179	12,179	35470,89252
30	54	432	0,065632	65,632	6582,155046

3. Pengujian *Throughput* dan *Delay* Hari 3 Sabtu 6 Januari

No.	<i>Length</i>	<i>bits</i>	<i>Delay</i>	ms	<i>Throughput</i>
1	54	432	0,057119	57,119	7563,157618
2	54	432	0,003801	3,801	113654,3015
3	54	432	0,058906	58,906	7333,718127
4	54	432	0,050942	50,942	8480,232421
5	54	432	0,01771	17,71	24392,99831
6	54	432	0,049702	49,702	8691,803147
7	54	432	0,051563	51,563	8378,100576
8	54	432	0,00764	7,64	56544,50262
9	54	432	0,06187	61,87	6982,382415
10	54	432	0,108068	108,068	3997,483066
11	54	432	0,010916	10,916	39574,93587
12	54	432	0,059655	59,655	7241,639427
13	54	432	0,05481	54,81	7881,773399
14	54	432	0,003952	3,952	109311,7409
15	54	432	0,060759	60,759	7110,057769
16	54	432	0,063747	63,747	6776,789496

17	54	432	0,007791	7,791	55448,59453
18	54	432	0,059629	59,629	7244,796995
19	54	432	0,050046	50,046	8632,058506
20	54	432	0,014728	14,728	29331,88485
21	54	432	0,055746	55,746	7749,434937
22	54	432	0,047184	47,184	9155,645982
23	54	432	0,007765	7,765	55634,25628
24	54	432	0,060297	60,297	7164,535549
25	54	432	0,054455	54,455	7933,155817
26	54	432	0,009337	9,337	46267,53775
27	54	432	0,063571	63,571	6795,551431
28	54	432	0,074143	74,143	5826,578369
29	54	432	0,010559	10,559	40912,96524
30	54	432	0,052071	52,071	8296,364579

4. Pengujian *Throughput* dan *Delay* Hari 4 Minggu 7 Januari

No.	<i>Length</i>	<i>bits</i>	<i>Delay</i>	ms	<i>Throughput</i>
1	54	432	0,006662	6,662	64845,39177
2	54	432	0,064252	64,252	6723,526116
3	54	432	0,052795	52,795	8182,593049
4	54	432	0,005489	5,489	78702,86027
5	54	432	0,060018	60,018	7197,840648
6	54	432	0,4731	473,1	913,126189
7	54	432	0,013482	13,482	32042,72363
8	54	432	0,048089	48,089	8983,343384
9	54	432	0,079475	79,475	5435,671595
10	54	432	0,011904	11,904	36290,32258
11	54	432	0,059593	59,593	7249,173561
12	54	432	0,072236	72,236	5980,397586
13	54	432	0,010603	10,603	40743,18589

14	54	432	0,056064	56,064	7705,479452
15	54	432	0,073618	73,618	5868,130077
16	54	432	0,005437	5,437	79455,58212
17	54	432	0,059199	59,199	7297,420565
18	54	432	0,068601	68,601	6297,284296
19	54	432	0,009337	9,337	46267,53775
20	54	432	0,059818	59,818	7221,90645
21	54	432	0,074686	74,686	5784,216587
22	54	432	0,004405	4,405	98070,37457
23	54	432	0,05604	56,04	7708,779443
24	54	432	0,07671	76,71	5631,599531
25	54	432	0,005737	5,737	75300,6798
26	54	432	0,056346	56,346	7666,915132
27	54	432	0,051501	51,501	8388,186637
28	54	432	0,00721	7,21	59916,78225
29	54	432	0,060385	60,385	7154,09456
30	54	432	0,070662	70,662	6113,611276

5. Pengujian *Throughput* dan *Delay* Hari 5 Senin 8 Januari

No.	<i>Length</i>	<i>bits</i>	<i>Delay</i>	ms	<i>Throughput</i>
1	54	432	0,011838	11,838	36492,65079
2	54	432	0,069424	69,424	6222,631943
3	54	432	0,01233	12,33	35036,49635
4	54	432	0,059425	59,425	7269,667648
5	54	432	0,005972	5,972	72337,57535
6	54	432	0,074032	74,032	5835,314459
7	54	432	0,002667	2,667	161979,7525
8	54	432	0,000949	0,949	455216,0169
9	54	432	0,078872	78,872	5477,228928
10	54	432	0,00509	5,09	84872,29862

11	54	432	0,072247	72,247	5979,487038
12	54	432	0,849933	849,933	508,2753582
13	54	432	0,013029	13,029	33156,80405
14	54	432	0,059962	59,962	7204,56289
15	54	432	0,005581	5,581	77405,48289
16	54	432	0,066293	66,293	6516,525123
17	54	432	0,014051	14,051	30745,14269
18	54	432	0,042554	42,554	10151,80712
19	54	432	0,229215	229,215	1884,69341
20	54	432	0,005942	5,942	72702,79367
21	54	432	0,057108	57,108	7564,614415
22	54	432	0,442753	442,753	975,713321
23	54	432	0,009655	9,655	44743,65614
24	54	432	0,06545	65,45	6600,458365
25	54	432	0,073561	73,561	5872,677098
26	54	432	0,001664	1,664	259615,3846
27	54	432	0,080011	80,011	5399,257602
28	54	432	0,05097	50,97	8475,573867
29	54	432	0,001852	1,852	233261,3391
30	54	432	0,063772	63,772	6774,132848

6. Pengujian *Throughput* dan *Delay* Hari 6 Selasa 9 Januari

No.	<i>Length</i>	<i>bits</i>	<i>Delay</i>	ms	<i>Throughput</i>
1	54	432	0,00854	8,54	50585,48009
2	54	432	0,052164	52,164	8281,573499
3	54	432	0,053263	53,263	8110,69598
4	54	432	0,008792	8,792	49135,5778
5	54	432	0,068013	68,013	6351,726876
6	54	432	0,069079	69,079	6253,709521

7	54	432	0,009768	9,768	44226,04423
8	54	432	0,062624	62,624	6898,313746
9	54	432	0,071996	71,996	6000,333352
10	54	432	0,006866	6,866	62918,72997
11	54	432	0,080061	80,061	5395,885637
12	54	432	0,052916	52,916	8163,88238
13	54	432	0,007474	7,474	57800,37463
14	54	432	0,063859	63,859	6764,903929
15	54	432	0,072443	72,443	5963,309084
16	54	432	0,005357	5,357	80642,15046
17	54	432	0,060016	60,016	7198,080512
18	54	432	0,063805	63,805	6770,629261
19	54	432	0,007289	7,289	59267,38922
20	54	432	0,070193	70,193	6154,459846
21	54	432	0,08817	88,17	4899,625723
22	54	432	0,005311	5,311	81340,61382
23	54	432	0,063622	63,622	6790,104052
24	54	432	0,054206	54,206	7969,597462
25	54	432	0,005071	5,071	85190,2978
26	54	432	0,060095	60,095	7188,618021
27	54	432	0,155579	155,579	2776,72437
28	54	432	0,011277	11,277	38308,06065
29	54	432	0,068149	68,149	6339,051197
30	54	432	0,108969	108,969	3964,430251

7. Pengujian *Throughput* dan *Delay* Hari 7 Rabu 10 Januari

NO	<i>Length</i>	<i>bits</i>	<i>Delay</i>	ms	<i>Throughput</i>
1	54	432	0,008688	8,688	49723,75691
2	54	432	0,052527	52,527	8224,341767
3	54	432	0,066125	66,125	6533,081285

4	54	432	0,006257	6,257	69042,67221
5	54	432	0,065413	65,413	6604,191827
6	54	432	0,06979	69,79	6189,998567
7	54	432	0,004459	4,459	96882,70913
8	54	432	0,06324	63,24	6831,119545
9	54	432	0,068152	68,152	6338,772156
10	54	432	0,006665	6,665	64816,20405
11	54	432	0,060074	60,074	7191,130939
12	54	432	0,074776	74,776	5777,254734
13	54	432	0,00509	5,09	84872,29862
14	54	432	0,05962	59,62	7245,890641
15	54	432	0,073671	73,671	5863,908458
16	54	432	0,009099	9,099	47477,74481
17	54	432	0,079829	79,829	5411,567225
18	54	432	0,070094	70,094	6163,152338
19	54	432	0,0083	8,3	52048,19277
20	54	432	0,062879	62,879	6870,338269
21	54	432	0,15302	153,02	2823,160371
22	54	432	0,011361	11,361	38024,82176
23	54	432	0,067082	67,082	6439,87955
24	54	432	0,072958	72,958	5921,214946
25	54	432	0,011413	11,413	37851,57277
26	54	432	0,057916	57,916	7459,078666
27	54	432	0,000427	0,427	1011709,602
28	54	432	0,075841	75,841	5696,127424
29	54	432	0,067866	67,866	6365,484926
30	54	432	0,002902	2,902	148862,8532

8. Pengujian *Throughput* dan *Delay* Hari 8 Kamis 11 Januari

NO	Length	bits	Delay	ms	Throughput
1	54	432	0,060108	60,108	7187,063286
2	54	432	0,9154	915,4	471,9248416
3	54	432	0,011304	11,304	38216,56051
4	54	432	0,060259	60,259	7169,053585
5	54	432	0,841056	841,056	513,6399954
6	54	432	0,010087	10,087	42827,40161
7	54	432	0,067446	67,446	6405,124099
8	54	432	0,128567	128,567	3360,115737
9	54	432	0,008347	8,347	51755,1216
10	54	432	0,060558	60,558	7133,65699
11	54	432	0,071167	71,167	6070,229179
12	54	432	0,007943	7,943	54387,51102
13	54	432	0,056143	56,143	7694,636909
14	54	432	0,318066	318,066	1358,208674
15	54	432	0,010403	10,403	41526,48275
16	54	432	0,050505	50,505	8553,608554
17	54	432	0,481333	481,333	897,5075468
18	54	432	0,01066	10,66	40525,32833
19	54	432	0,066097	66,097	6535,848828
20	54	432	0,979575	979,575	441,00758
21	54	432	0,006812	6,812	63417,4985
22	54	432	0,072004	72,004	5999,666685
23	54	432	0,082397	82,397	5242,909329
24	54	432	0,01199	11,99	36030,02502
25	54	432	0,06807	68,07	6346,408109
26	54	432	0,007944	7,944	54380,66465
27	54	432	0,052342	52,342	8253,410263

28	54	432	0,337811	337,811	1278,82159
29	54	432	0,012515	12,515	34518,57771
30	54	432	0,048247	48,247	8953,924596

[Halaman ini sengaja dikosongkan]

RIWAYAT HIDUP



Muhammad Alfian Yasir, lahir di Jakarta tanggal 01 Mei 1995, Pendidikan dasar terakhir di SDN Cipayung 01 Jakarta Timur pada tahun 2007, lalu melanjutkan Pendidikan di SMPN 09 Jakarta Timur dan lulus pada tahun 2010, setelah itu melanjutkan Pendidikan menengah atas di SMAN 48 Jakarta Timur dan lulus pada tahun 2013. Setelah lulus Pendidikan menengah atas, lalu melanjutkan Pendidikan strata satu (S1) departemen Teknik Elektro di Institut Teknologi Sepuluh Nopember Surabaya pada tahun 2013. Penulis dapat dihubungi melalui kontak 081364334204 atau melalui email ini: muhammadalfian.yasir083@gmail.com